University of Colorado
Department of Aerospace Engineering Sciences
ASEN 4018

Project Final Report (PFR)

P4LO - Positioning For Lunar Operations

Project Customers

Name:	Scott Palo	Name:	Nicholas Rainville
Email:	Scott.Palo@colorado.edu	Email:	Nicholas.Rainville@colorado.edu

Project Advisor

Name:	Jade Morton
Email:	jade.morton@colorado.edu

Team Members

Name: Sam D'Souza Email: sads9574@colorado.edu Phone: (303) 956-4085	Name: Ponder Stine Email: post1579@colorado.edu Phone: (720) 588-6563
Name: Alexander Lowry Email: allo5581@colorado.edu Phone: (979) 267-1188	Name: Dawson Weis Email: dawe5235@colorado.edu Phone: (636) 222-3046
Name: Ruben Hinojosa Torres Email: ruhi9621@colorado.edu Phone: NA	Name: Nathan Jager Email: naja2101@gmail.com Phone: (720) 579-2399
Name: Brendan Palmer Email: Brendan.Palmer@colorado.edu Phone: (702) 595-7846	Name: Ventura Morales Email: vemo8309@colorado.edu Phone: (720) 630-9565
Name: Fernando Palafox Email: Fernando.Palafox@colorado.edu Phone: (720) 666-2131	Name: Ian Thomas Email: Ian.Thomas-1@colorado.edu Phone: (831) 345-6514

Contents

List of Figures

List of Tables

1 Acronyms & Nomenclature

GPS Global Positioning System
$GPSDO$ GPS-Disciplined-Oscillator
LPS Local Positioning System
RF Radio frequency
$BPSK$ Binary phase-shift keying
$CDMA$ Code-division multiple access
SMS Short message service (text message)
JPL NASA's Jet Propulsion Laboratory
CU The University of Colorado, Boulder
σ Standard deviation
$CONOPS$ Concept of operations
$HDOP$ Horizontal Dilution of Position
SNR Signal-to-noise ratio
AM Amplitude modulation
FM Frequency modulation
GUI Graphical user interface
GNU GNU's not unix
$TDOA$ Time difference of arrival
CPU Central Processing Unit

2 Project Purpose

Author: Alex Lowry, Dawson Weis

Figure 1: LunaNet Mission [2]

The year is 2040. We have a base for re-fueling on the moon and have an established civilization beginning on Mars. We are beginning to plan missions to venture deeper into our solar system. However, as we continue to expand and explore further into space, astronauts need ways to stay connected. On the Moon, astronauts need to be able to know their position and where they are navigating to in order to explore safely. They need to be able to receive alerts in real-time, warnings for unprecedented space weather, such as solar flares, and incoming asteroids. Eventually, there may be a need for a networking system that expands to the entire solar system. The first step to achieving this is the LunaNet Program [1]. This program calls for handheld communication with astronauts on the lunar surface; however no specific groundwork has been mapped to achieve this goal.

Hence, this project, named P4LO (Positioning for Lunar Operations), focuses on developing a proof of concept system designed to demonstrate the ability to bring a Local Positioning System (LPS) similar to GPS on Earth, and SMS-like reception and transmission to the moon through a network of software-defined radios (SDR). While full scale testing cannot be done through the scope of this project, the team will focus on scaling-down and localizing the design solution such that, for practical purposes, the feasibility of the proof of concept can be developed. Furthermore, the infrastructure of the system will demonstrate a prototype reflecting the fundamental components required for a potential lunar mission. The team will also conduct risk mitigation by testing capabilities of both the hardware and software. This will reduce risk for future JPL/LunaNet work, allowing those teams to avoid obstacles along the way.

This project is significantly different than any other current communications and navigation system. For this system, potential satellites would be at lower orbits than Earth-orbiting satellites, causing their velocity to increase by a large amount. This makes tracking more difficult, and increases uncertainty in ranging. Because of these issues, traditional GPS systems would not be ideal for implementing our communication and positioning system on the lunar surface. Therefore the team has been tasked with designing a positioning and communication system that will work in a localized area that can scale to meet performance needs of the LunaNet mission.

3 Project Objectives and Functional Requirements

Author: Nathan Jager, Alex Lowry, Sam D'Souza

The objectives for this project are to design a prototype handheld communication device that demonstrates a path to a viable communication and navigation solution network to be implemented on the lunar surface. Through the development of this prototype, our team will be providing a risk reduction and preliminary analysis for the future implementation of the customer's contribution in the LunaNet system. The team will primarily be focused on building upon a previous LimeSDR prototype provided by CU graduate students to demonstrate the communication link between astronauts on the lunar surface and orbiting satellites. Here, the LimeSDR is acting as our software defined radio, a communication system in which traditional hardware components have been transitioned to a software implementation. Since testing on the lunar surface is out of the scope of the project, the team will instead focus primarily on building a LimeSDR-based prototype to demonstrate the communication link between users in a localized region covered by surrounding pseudolites. Pseudolites are defined to be a variation of traditional satellites. These devices include all the traditional components of satellites in which they can transmit and receive radio frequency signals and provide navigation solution for their covered region while operating on the ground. In order to do this, the team will analyze and optimize a pseudolite configuration that satisfies our customer positioning requirements of sub 10 meter positioning and 30 nanosecond $1-\sigma$ transfer time.

This PFR will focus on two integral paths of the overall mission: development of the communication link between the user and the pseudolites, and a simplified model of the positioning system.

3.1 Levels of Success

Author: Ponder Stine, Alex Lowry

Level	Title	Requirements
1	Hardware/Software Interfacing	Ubuntu environment and dependencies for GNU Radio and LimeSDR
2	Wireless Transmission	Completing a simple Rx/Tx test, Simple communications test
3	Acquisition	Identify signal/PRN codes for separate transmitting devices
4	Positioning	Ranges from acquisition results integrated with positioning algorithm
5	Data Transmission	Transmit .txt file over the air at specified data rate.

Table 1: Levels of Success

In Table 1 above, the levels of success for our project have been outlined. The first level focuses on the actual hardware and software interfacing, as this is crucial to being able to test with transmission and acquisition. This requires the Ubuntu environment to be set up correctly along with the interfacing of the GNU Radio and the LimeSDR. The second level of success focuses on the communication aspect of the project, where the team attempts to perform wireless transmission. The requirement for this is to complete a simple receiver and transmitter test, where signals are sent to and from the pseudolites. The third level of success focuses on signal acquisition, where the team is attempting to receive and identify signals from the transmitters in order to determine ranges for the position calculations in the positioning algorithm, as outlined in level of success number 4. Finally, our last level of success is data transmission, where the team attempts to transmit a text file at a specified data rate. This is the most difficult task of the project and is the final test in determining whether our communication and positioning solutions are viable.

3.2 CONOPS

Author: Nathan Jager, Alex Lowry, Ruben Hinojosa Torres

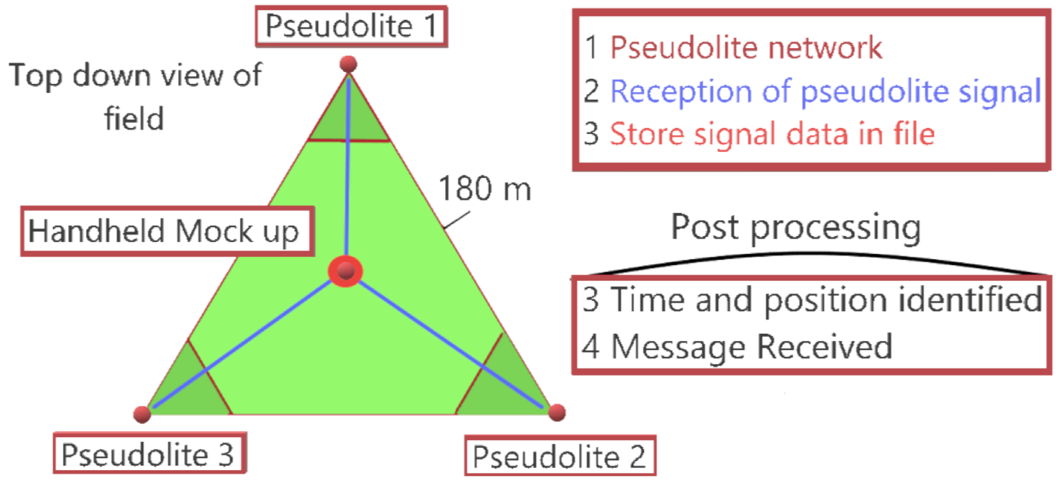

Figure 2: Concept of Operations

The team's mission consists of two sub-projects: a low-rate communication prototype and a ground-based navigation system. The communication prototype sub-project is intended to facilitate SMS-like communication between our three positioning and communication towers (pseduolites), which are set-up in a equilateral triangular configuration. Our receiver system is depicted in the diagram as the handheld mock up, which is located in the center of the region. The ground-based navigation sub-project will consist of a GPS-like system with positioning capabilities that will allow for our receiver to determine their position in our testing region. The team will use software defined radios (SDR), specifically the customer required LimeSDR, to mimic the proposed communications system and test the functionality and accuracy of the positioning software. The customer defined requirements are to be able to have positioning data accurate to within 10 meters, time data accurate to within 30 nanoseconds, receive signals at 2.48 GHz, transmit signals at 2.4 GHz, while transmitting data at at least 200 bits per second.

3.3 Functional Block Diagram

Author: Alex Lowry

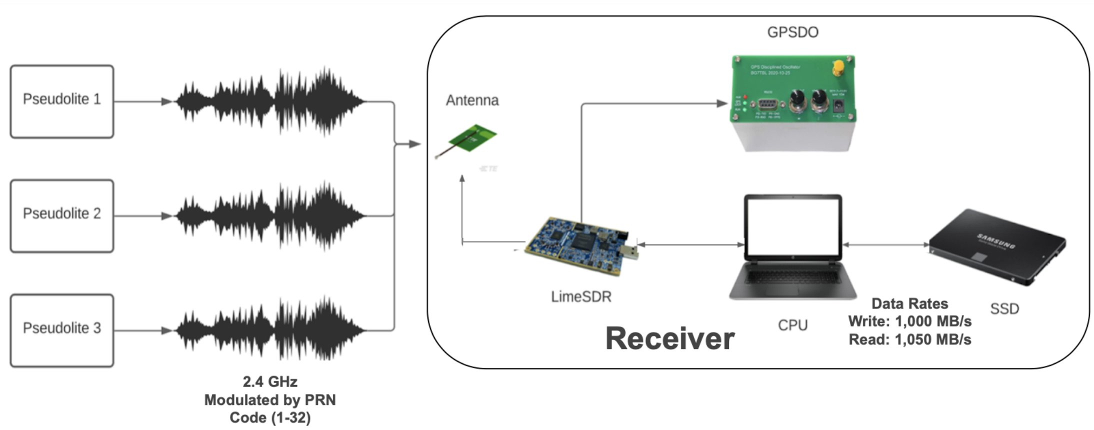

Figure 3: Functional Block Diagram for Whole System

The functional block diagram above shows an overview of the overall system. On the left side of the diagram, there are three pseudolites which correlate to the pseudolites shown in Figure 2. These three pseudolites handle navigation determination of our system. A functional block diagram of the pseudolite itself and an in-depth description of its components can be found in Figure 4 in Section 4.2, 'Pseudolite Design'.

Referring back to Figure 3, three of these pseudolites can be seen on the left transmitting signals to the receiver on the right. It should be noted that the receiver closely resembles the functional block diagram of the pseudolite in Figure 4. This is accurate, except the antenna is receiving signals rather then sending them. The receiver system then utilizes the LimeSDR, where it undergoes analog down conversion to process the signal to an intermediate frequency. From here, it can be digitally sampled for signal processing. In Figure 4 below, this signal processing is highlighted.

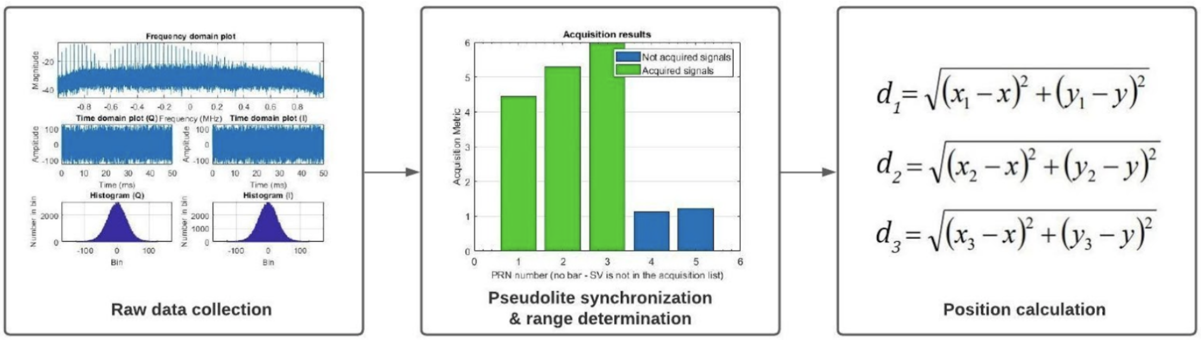

Figure 4: Functional Block Diagram for Signal Processing

When the PRN codes are transmitted from the pseudolites, the software receives and samples the raw data and the signal. From that, pseudolite synchronization and range determination is performed, where via a MATLAB acquisition script, the software can determine whether or not the signals were acquired. Finally, using the ranges from the pseudolite to the receiver that was found, position calculations can be done using the equations on the right of the figure, where the user's location in the x, y, and z coordinates can be determined. This process will be explained in more detail later on in this report.

3.4 Functional Requirements

Author: Sam D'Souza

Our functional requirements (**FR**) for our pseudolite network are as follows:

FR 1: The system must operate under a scalable Local Positioning System (LPS).

Rational: Functional Requirement 1 details the overarching goal of this project, which is to be able to develop a positioning network that can scale to provide navigation over a variable amount of space. Because the development of this project fits into the much larger LunaNet objective, the precise scale of operations is still being determined. Thus, our system must provide a positioning system that can scale to meet overall mission objectives down the road.

FR 2: The system will provide two-way SMS-like texting capabilities

Rational: Functional Requirement 2 details the second most important goal of this project, which is to provide means to communicate with our systems on the lunar surface. Whether this is communication between two astronauts, astronauts communicating to home, or electronic systems communicating with each other, two-way communication between entities within our pseudolite network is critical in providing mission success.

FR 3: The system will be able to provide an architecture for a navigation solution receiver with a 10 meter position accuracy and a 30 nanosecond 1-σ transfer time

Rational: Functional Requirement 3 details a strict customer need in order to meet mission objectives. Therefore, our system must meet this 10 meter positioning accuracy and 30 nanosecond 1-σ transfer time requirement in order to satisfy our customers need.

FR 4: The system will transmit data at 2.4GHz carrier frequency

Rational: Functional Requirement 4 details another customer constraint. Due to the limited frequency bands available for lunar communications, our communication channel is constricted to operate between 2.4 - 2.48 GHz. The team's design choice for transmission frequency is 2.4 GHz.

FR 5: The system will receive data at 2.48GHz carrier frequency

Rational: Functional Requirement 5 details the same customer constraint. The team's design choice for receive frequency is 2.48 GHz.

FR 6: The system framework must be extendable to 170 users communicating simultaneously over the lunar region

Rational: Functional Requirement 6 details another customer constraint, namely 170 users must be able to communicate simultaneously over the lunar region. This functional requirement dictates our communication architecture and is therefore the backbone to our communication channel.

FR 7: The system's communication links must have a channel bandwidth of no more than 1 MHz

Rational: Functional Requirement 7 details another customer constraint within our communication link. Therefore, our system must have a channel bandwidth of no more than 1 MHz.

FR 8: Communication must operate at a data rate of at least 200 bits per second

Rational: Functional Requirement 8 was derived off our customer requirements. The customer requirement was to provide a data rate that is feasible for SMS-like communication. The team therefore deduced that a data rate of 200 bits per second is sufficient for traditional SMS-like communication.

4 Final Design

This section is composed of two parts, the requirements flow-down and the final design description. In the requirements flow-down, the team describes how functional requirements were turned into detailed design requirement. Additionally, rationale for each major requirement is also provided. In the final design description, specifics on the whole system are given including pseudolite design, transmitted signal design, pseudolite geometry and clocks and GPS-disciplined Oscillators. The conceptual design alternatives that were considered for this design can be found in Appendix 12.

4.1 Requirements Flow-down

Author: Alex Lowry, Dawson Weis

FR 1: The system must operate under a scalable Local Positioning System (LPS).

DR 1.1: Pseudolites will be used in place of traditional satellites (like GPS) to provide communication links and positioning solutions.

Rational: Pseudolites are ground-based and have similar components to that of a traditional satellite. They are easy to design and set-up, and will fit within the team's financial budget. These pseudolites will allow the team to practically test their solution under scalable conditions as it will closely replicate a system on the lunar surface.

DR 1.2: The system will operate using the LimeSDR.
Rational: In order to create the Local Positioning System, the team must have the hardware and system setup necessary to verify transmission and reception of signals. The LimeSDR provides all of the capabilities required to transmit and receive signals, as well as interface with other hardware.

FR 2: The system will provide two-way SMS-like texting capabilities

DR 2.1: Device must demonstrate wireless transmission and reception of data.
Rationale: The focus of the project is to provide communication and positioning on the Lunar surface. A functioning part of this goal is to be able to receive SMS-like messages and LPS data, as well as transmit SMS-like messages. Conducting transmission and reception wirelessly allows users to communicate from long distances away from other users or pseudolites.

DR 2.2: Messages must be processed and displayed on a screen for a user to read and analyze. Also the user must be able to send a message of their own.
Rationale: Since the project requires viable communication between users, it is important that they can read and analyze the messages that they send and receive.

FR 3: The system will be able to provide an architecture for a navigation solution receiver with a 10 meter position accuracy and a 30 nanosecond 1-σ transfer time

DR 3.1: Devices must receive signals at SNR of 23.52 dB
Rationale: Signal-to-noise ratio (SNR), along with bandwidth (B), are intimately related with desired range accuracy by the following formula [6]:

$$\delta R = \frac{c_0}{2B\sqrt{2SNR}}$$

We have a desired position accuracy $\delta R = 10$ meters, the speed of light in a vacuum $c_0 = 3 \times 10^8$ m/s, and a bandwidth $B = 1$ MHz, from FR7. This yields an SNR of 225, or 23.52 dB.

DR 3.2: Provide at least 90% coverage in the pseudolite region.
Rationale: The poles are of highest interest on the Moon, and so it can be assumed that astronauts will be spending most of their time at these locations. Hence, to ensure the astronauts consistently know their location, it is important that the team's solution covers 90% of the region of interest. In our testing environment, this will be scaled down accordingly.

FR 4: The system will transmit data at 2.4GHz carrier frequency

DR 4.1: Demonstrate uplink (receiver to pseudolite) transmission at 2.4GHz
Rationale: Communication needs to occur at an agreed upon frequency for transmission and demodulation to occur. For lunar communications, the communication channel is limited to operations between 2.4 - 2.48GHz. Thus, the customer requirement is 2.4GHz for transmission.

FR 5: The system will receive data at 2.48GHz carrier frequency

DR 5.1: Demonstrate downlink (pseudolite to receiver) transmission at 2.48GHz
Rationale: Communication needs to occur at an agreed upon frequency for transmission and demodulation to occur. For lunar communications, the communication channel is limited to operations between 2.4 - 2.48GHz. Thus, the customer requirement is 2.48GHz for reception.

FR 6: The system framework must be extendable to 170 users communicating simultaneously over the lunar region

DR 6.1: Demonstrate simultaneous communication with 2 users within testing region.
Rationale: A customer given requirement is that the overall system must service 170 users within the lunar region. This allows for the system to provide practical value on the Lunar surface for multiple astronauts. However, since the team cannot practically test this number of users, the solution must be scaled down. For each individual cell in the pseudolite region, at least 2 users must be able to communicate simultaneously.

DR 6.2: Scale down from 2.4km between each pseudolite to 180 meters between each pseudolite.
Rationale: 2.4km is the horizon distance from a user at a height of 2 meters on the lunar surface. Thus, it will be assumed that pseudolites would be positioned from each other at a maximum of this distance on the moon. However, since this distance is not testable for the team, the solution must be scaled down while also providing sufficient evidence of feasibility such that it can be extendable to the lunar set-up with 170 users.

FR 7: The system's communication links must have a channel bandwidth of no more than 1 MHz

DR 7.1: Uplink and downlink channels must have a maximum bandwidth of 1 MHz
Rationale: Various international organizations, such as the International Telecommunications Union (ITU), have placed restrictions on communication links in different jurisdictions. As a result, the customer requested that our signal take up no more than 1 MHz of bandwidth.

FR 8: Communication must operate at a data rate of at least 200 bits per second (per user)

DR 8.1: User transmits and receives communications data at a data rate of at least 200 bits per second.
Rationale: In order to send and receive communication data to pseudolites in a timely manner, a 200 bits/s per user data rate is required to accommodate both SMS-like communication and LPS data simultaneously

To satisfy the functional requirements listed above, the team created a three pseudolite - one receiver system. Each pseudolite transmits a signal which is acquired by the lone receiver. The pseudolite structure is shown below.

4.2 Pseudolite Design

Author: Alex Lowry, Ruben Torres, Brendan Palmer
Each pseudolite consists of the same 5 components. A working block diagram of these components can be seen in Figure 5.

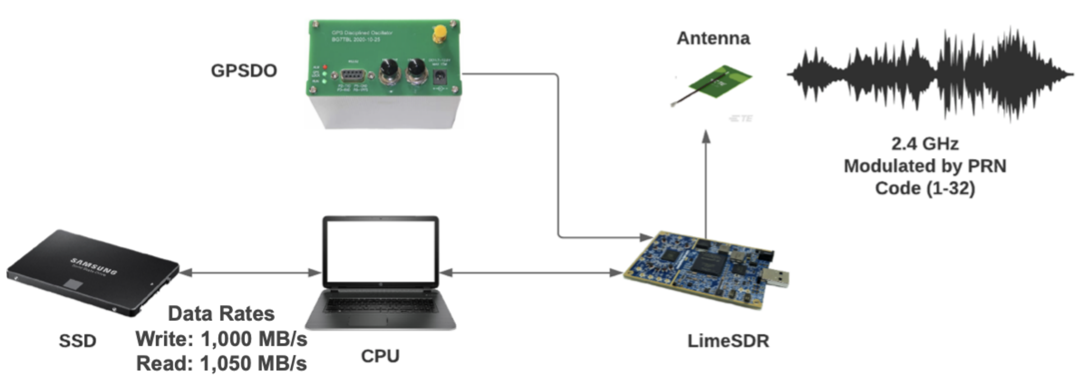

Figure 5: Functional Block Diagram for a Pseudolite

From the figure, the pseudolites break down as follows: a solid state drive running Ubuntu Linux OS, a CPU, a LimeSDR, a GPS Disciplined Oscillator (GPSDO), and a linearly polarized TE Connectivity patch antenna. Each of these components is used to satisfy different functional requirements. First, a LimeSDR is used to satisfy our customer requirement of using a LimeSDR to create a lunar communication and positioning system. To utilize the LimeSDR, the team used GNURadio to execute transmission and reception scripts. To unify the pseudolite models and satisfy the read/write speed requirements, the team attached an SSD running Linux OS with GNURadio installed to a CPU. This allowed each pseudolite to have the same software packages installed and utilize the same scripts. The CPU alone was used to post-process the data using MATLAB. The GPSDO was used to overwrite the relatively poor clock on the LimeSDR in order to minimize clock drift and provide a stable oscillator, which allowed for higher quality transmission and reception of signals. Finally, the signal is modulated onto a 2.4 GHz carrier wave using GNURadio and sent to the receiver via a linearly polarized patch antenna.

4.3 Signal Architecture

Author: Ian Thomas, Fernando Palafox, Brendan Palmer

For this project, three pseudolites are required to simultaneously send signals to a receiver while transmitting at a single frequency. In order to accomplish this without unwanted interference, a process called multiplexing must be performed. While there are many multiplexing schemes, the team chose pseudo-random Code-Division-Multiple-Access(CDMA). In CDMA, a unique pseudo-random bit stream, or code, is assigned to each transmitter. Data bits at a rate of 1 kbps are first modulated onto the spreading code, which is a maximal length Gold code at a rate of 1.023 MHz. Since the codes are 1023 bits long, each data bit is modulated onto one full spreading code. The codes are then modulated onto a carrier frequency at 2.4 GHz and amplified before being transmitted. The process of modulating a CDMA code on a signal at baseband is shown in Figure 6 below.

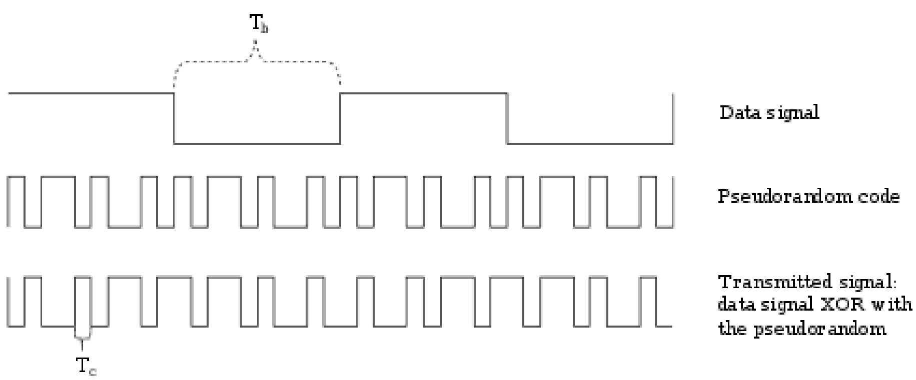

Figure 6: CDMA Multiplexing

Once the signal reaches the receiver, it is digitally sampled at a rate of 20 Msps and downconverted to baseband. The signal then undergoes a process called acquisition, in which a replica pseudorandom code is generated and compared with the baseband signal via cross-correlation. This process will indicate whether or not that particular code is present in the signal, and determine the delay of the code present in the signal relative to the reference code generated by the receiver. This code delay information is then used to extract the data bits and provides a range measurement with a resolution determined by the sample rate.

4.4 Pseudolite Geometry and Testing Setup

Author: Fernando Palafox

For the positioning system, the team designed the pseudolite geometry seen in Figure 7. This geometry consists of three pseudolites (represented as red diamonds) arranged in an equilateral triangle with a side length of 0.18km. The geometry was chosen in order to satisfy the number of pseudolites required to calculate a position (at least 3 are needed if one wishes to calculate the 3 unknowns: x, y and clock bias). Note that the scale of this system was chosen for practical purposes - in the real-life setup, the triangle side lengths will be closer to 2.4km (the visible horizon on the Moon). The triangular shape of the geometry means that expanding the system by adding adjacent cells is very easy. Because of this reason, only locations within the triangle will be tested, since anything outside will be assumed to be taken care of by adjacent cells. The color gradient seen in the inside of the triangle is an indication of Horizontal Dilution of Precision (HDOP) which is a measure of uncertainty in a position solution. This number is also a multiplier to the overall error of the system, therefore the closer it is to 1, the better. As seen in the plot, the average is 1.3 which is very good. Further details on the overall distribution of HDOP within the triangle can be found in the Appendix 11.5. Another important feature of the geometry are the cutouts seen in the corners close to each pseudolite. These cutouts represent the regions in which the near-far problem causes one pseudolite's signal to overpower the rest. These regions are not ideal for testing and will not be used for testing. Further details and a plot detailing the near-far calculations done can also be found in Appendix 11.4.

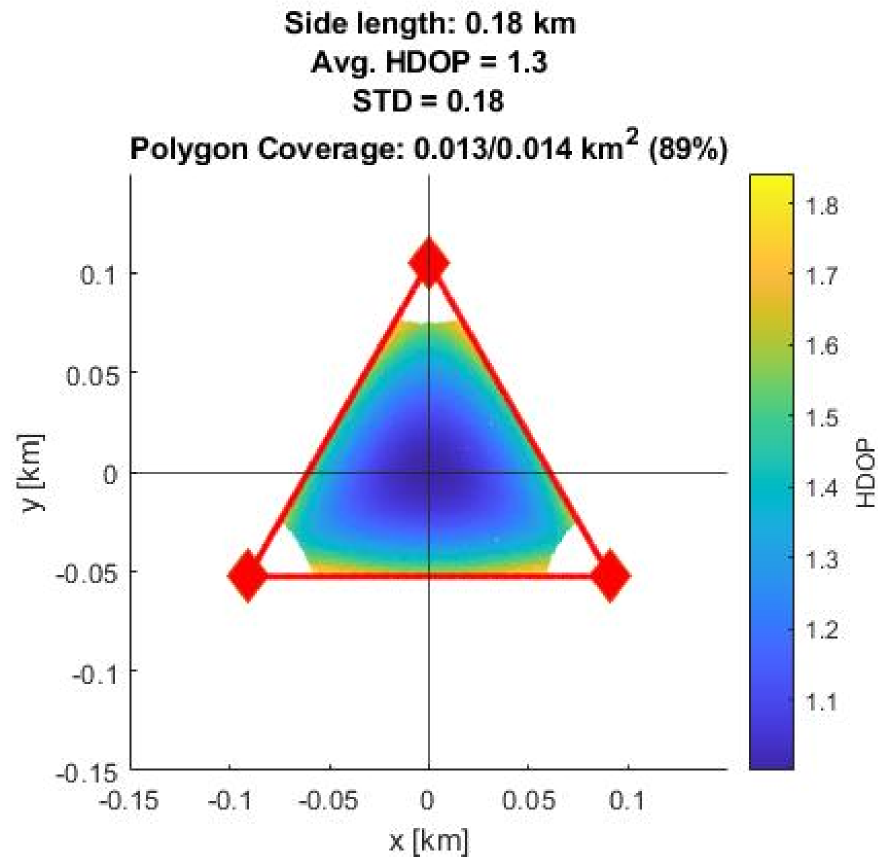

Figure 7: Pseudolite Geometry

4.5 Clocks and GPS-Disciplined Oscillators

Author: Alex Lowry, Fernando Palafox

Due the use of trilateration with one-way ranging as the chosen positioning algorithm and in order to satisfy positioning functional requirements (FR3), all pseudolite clocks must remain synchronized to within 30ns for the duration of a field test (around 2 hours). The clocks on board the LimeSDRs are temperature compensated crystal oscillators (also known as TCXOs), which drift at an average of 0.05ns, every second. This means that the threshold value of 30ns will be exceeded after only 10 minutes. In order to solve this problem, the team decided to override the pseudolite clock systems in a process known as GPS-disciplining. This process uses a GPS receiver outfitted to each pseudolite in order to compute receiver clock bias and therefore keep each clock synced with GPS time. Furthermore, the high-quality oscillators within the receiver can also be used to generate a higher-quality signal that will be easier to track by the receiver. Thus, a GPS-Disciplined Oscillator (GPSDO) was connected to each

LimeSDR of the transmitters, as was previously mentioned in the Final Design section of this report. This ensures higher quality signal transmission. The user receiver will also be using a GPSDO in order to use its high quality oscillator which will aid with signal acquisition and tracking.

5 Manufacturing

5.1 Purchased vs Constructed

Author: Dawson Weis, Nathan Jager

Team P4LO purchased several components to that were used in the construction of the pseudolite system. The components consisted of software defined radios, micro controllers, GPS disciplined oscillators, mobile power suppliers, antennas, etc. displayed in figure 8.

Component Purchased	Purpose
GPSDO	Increase transmitter/receiver clock stability
SSD	Store data, boot programs like GNU Radio through Ubuntu
Power Packs	Extend to outside testing for pseudolites (power GPS-DO, LimeSDR,etc)
Extra Adaptors/ Extenders	USB-Extender cables and adaptors for pseudolites (needed for accommodating pseudolite housing)
Raspberry PI	Pi’s were initially purchased to conduct code computation

Figure 8: Purchased Components

After buying all of the hardware above, the team still had some manufacturing to do themselves. The pseudolites had to be built, as well as the software scripts developed to run the communications protocol. Both of these processes are outlined below.

5.2 Pseudolite Construction

Author: Dawson Weis, Nathan Jager

In order to create stability and consistency in our testing scenario, the team needed to create a sufficient housing to hold all of the hardware necessary for each pseudolite. This removed the process of unplugging and replugging every connection into each other every time that tests were conducted. This decreased testing setup time, as well as decreased wear-and-tear on the hardware. In addition, improved portability allowed for easier transportation of all hardware components. Finally, the pseudolite housing allowed for consistent pointing of the antenna. Instead of holding the antenna manually, the antenna could be pointed in the same, nearly vertical, position each test. The physical construction of a working pseudolite can be seen below in Figure 9.

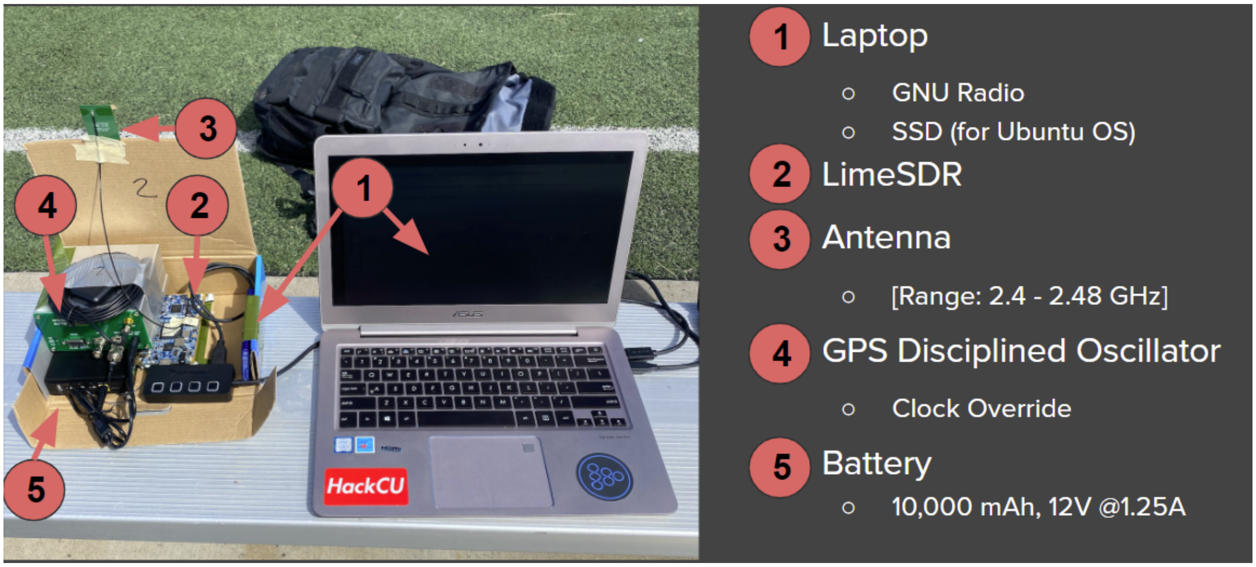

Figure 9: Physical Pseudolite, with Description of Elements

Earlier in the project cycle, the team had planned on using a 3-D printed housing. However, the CU labs did not have 3-D printers large enough to print a container that would hold all the parts, and outsourcing to 3-D printing companies was too expensive and time consuming. Thus the team decided on a cheap, effective design. Cardboard boxes were cut and used to house the pseudolites, which worked as intended and were free to make. Inside the housing, each component was either glued or taped to the box for restricting movement of components. In order to minimize size, some components such as the GPSDO and battery, were stacked and taped into a single piece. The antenna connection to the SDR was also taped down, in order to make sure it does not become unplugged. Finally, the antenna itself was taped to the top of the box, where it was able to maintain a fairly consistent pointing direction.

After constructing the pseudolites, the team were able to easily setup and complete tests. The open parts of the cardboard box worked well to allow access for plugging in USB connections during the testing. Figure 13 shows the usage of three pseudolites housing sitting next to each other for testing.

5.3 Software

Author: Ian Thomas

The team needed a processing architecture that could operate with the LimeSDR and support the necessary processing and data speeds required to send and receive CDMA signals.

5.3.1 Operating System and Hardware

The team opted for a laptop notebook running Ubuntu 20.04 off a Samsung T7 portable external solid state drive. The team chose the Linux environment because the vast majority of prior work on LimeSDR was done in Linux, with tools such as GNURadio and LimeSuiteGUI already built and packaged for installation for Linux. The rationale behind running the operating system of an external SSD was that the laptops of the team members already had Windows running, and instead of booting Linux alongside Windows on the laptops, the team could use SSDs with a portable Ubuntu 20.04 installation so complete, identical versions of the software could be installed and packaged in a single SSD. This enabled all the team members to have instant access to the full software environment without having to worry about setup and installation on their own computers, provided they had an SSD. The SSDs could also be kept with the pseudolite and would function properly with any Windows laptop.

The read/write speed of the SSD (1000 MBps) proved to be crucial during some tests which required a 20 MHz sample rate, as the team's previous choice of the LimeSDR could only push 30-40 MBps through a USB port and would not have allowed the team to record data at a rate of 20 MHz.

5.3.2 GNURadio Transmit Script

After the operating system was setup, the team installed a `GNURadio` version 3.8.2 alongside `LimeSuiteGui` and `gr-limesdr` via PPA. The GNURadio transmit script pointed to a binary file containing the bits of information to be sent as the least significant bit of 8-bit unsigned integers, and sampled them at the correct data rate before passing them to the LimeSDR for up-conversion and transmission. The following is a screenshot of the GNURadio transmit flowgraph:

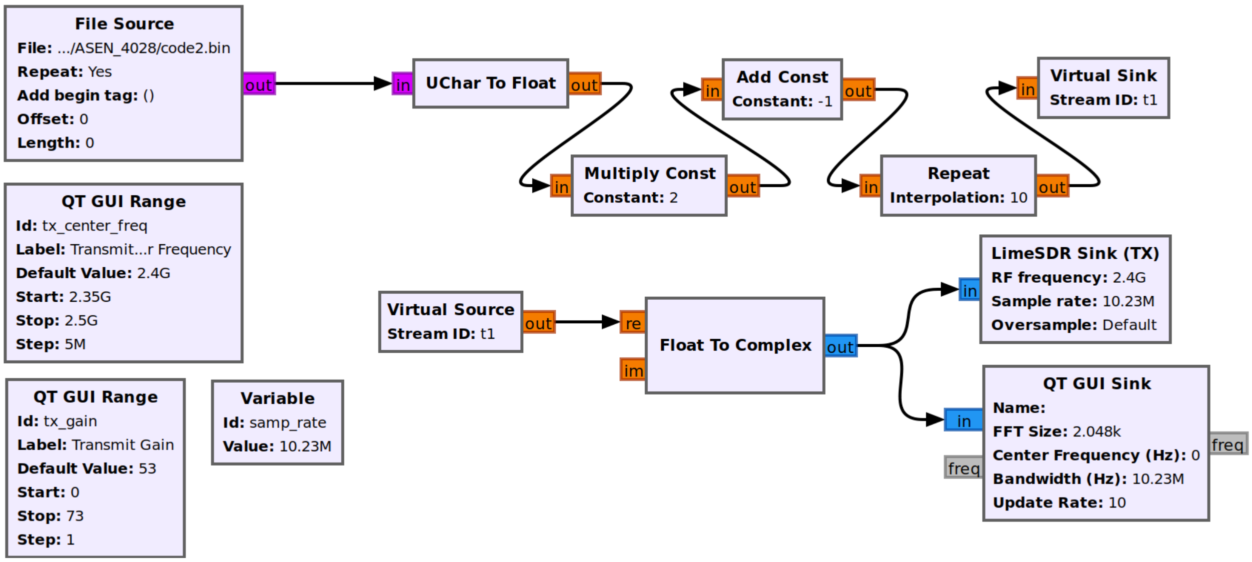

Figure 10: GNURadio Transmit Flowgraph

5.3.3 MATLAB Receive Script

The receiver script was initially implemented in GNURadio (recording the raw samples to a file) for post-processing in MATLAB, but there was a large overhead of constantly shifting large data files from the Linux environment to MATLAB running on Windows. There wasn't much documentation for getting the LimeSDR working with MATLAB in Windows, but the team was able to get it to work after downloading the `LimeSuiteGui` dynamic linked library (DLL) and carefully following the limited instructions for installation. This enabled the team to integrate the LimeSDR directly into the MATLAB processing script without any file transfer overhead.

5.4 Challenges and Lessons Learned

In the early stages of testing without the pseudolite housing, the team still had to unplug and replug each hardware connection each time testing was done. This potentially could have interfered with the antenna, as well as other hardware performance through eventual wear-and-tear of connections. As mentioned previously, the pseudolite housing is not what we had initially designed for. Originally the team was going to use the standard 3-D printing material PLA in order to model, and then 3-D print the housing. Due to the size limitations of the 3-D printers at CU, we could not print on campus. The team then looked at outsourcing the printing to a company, but the resulting cost, as well as the time it would take to get the model back, were both too large. If we had been able to get this 3-D printed housing, the project would have looked more professional, as well as had better consistency than the cardboard model.

From the manufacturing experience, the team learned a lot. Firstly, it is important to research the manufacturing process early in the process to get an idea of what we need, as well as how long it will take. We waited until February to reach out to CU about 3-D printing, so when we found out that the size limitations were an issue, we were already fast approaching testing. The team should have done this research in the fall.

On the data processing side, the team ran into some challenges running both the SSD and LimeSDR through one USB hub connected to one port on the computer. Depending on the type of USB port, some samples would get lost in this configuration. There were also some concerns that some USB ports were not able to supply enough power to the LimeSDR and the SSD. During testing, the team noticed that some computers were not able to transmit a signal with enough power and some computers would lose samples on the receiver side, but it took a few weeks to determine whether the issue was software or hardware related. A lot of time was spent debugging the transmit/receive scripts when the issue was really the hardware connections.

6 Verification and Validation

6.1 Levels of success

Author: Ruben Hinojosa Torres

Level	Title	Requirements	status
1	Hardware/Software Interfacing	Ubuntu environment and dependencies for GNU Radio and LimeSDR	Completed
2	Wireless Transmission	Completing a simple Rx/Tx test, Simple communications test	Demonstrated
3	Acquisition	Identify signal/PRN codes for separate transmitting devices	Demonstrated
4	Positioning	Ranges from acquisition results integrated with positioning algorithm	Capability in place
5	Data Transmission	Transmit .txt file over the air at specified data rate.	Capability in place

Table 2: Levels of Success

In Table 2, we can see what the team set out to do and what the team accomplished, as denoted by the status column. Our first level of success was concerning the hardware interface with the software setup required to get the development environment working. Considering many CU Aerospace teams had a hard time getting the GNU Radio software package to work, we established this as our base level of success. Our second level of success was to then test the installation with simple Rx to Tx port, over the air, transmission test. This was to ensure that the installation was successful and that we would be able to achieve communications between SDR's. The third level of success incorporated some of the communications aspects to the transmission. At this point we were applying our knowledge of BPSK signals and how to encode binary data into these signals. For the fourth level of success we are to take the acquisition results and measure a position solution with these using a positioning algorithm, more on this later in the report. For our final level of success we then extend all of these capabilities to transmit a message via the SDR's.

6.2 Test Description

Author: Ruben Hinojosa Torres

Four tests were created to satisfy our set of functional requirements. First, a basic communications test between our Tx and Rx channels, a test of the oscillator override, a short-range acquisition test for multiple transmitting devices, and finally the final long-range acquisition test.

6.2.1 Basic Communication Test

Our first test consisted of verifying basic communication capabilities between a single pseudolite and the receiver. The test consisted of a laptop running GNU Radio on the Ubuntu environment setup on an SSD and the LimeSDR with two antennas (one on a transmit port and the other on a receiving port.). A test script was then created for GNU Radio so that an FM audio file could be transmitted through the transmission port of the SDR to the receiving port of the same SDR. This simple test verified transmission and reception capabilities as well as a working LimeSDR environment. Below we see a picture (Fig. 11) taken of the test setup.

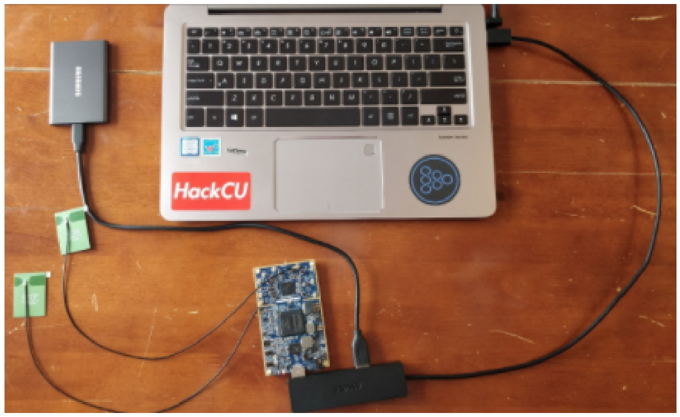

Figure 11: Basic Communication Test Setup

6.2.2 Oscillator Override

Our second test was a result of some analysis of the requirements that needed to be met by the system. As mentioned in the final design section, the on-board clock of the LimeSDR is not very accurate and does not have enough stability to provide measurements that can meet functional requirement 3, "to provide an architecture for a navigation solution receiver with a 10 meter position accuracy and a 30 nanosecond 1-σ transfer time." The team solved this issue by overriding the on-board clock with a GPS disciplined oscillator (GPSDO). The GPSDO is then used as a new clock source to allow higher quality transmission and reception. The test consisted of setting up a pseudolite as a GPS receiver and performing a series of position solutions. These position solutions were then compared to a known truth, and performance was assessed as a proxy for clock stability. Additionally, this test allowed the team to verify the functionality of the positioning algorithm. Below we see the specific GPSDO (Fig. 12) used in the receiver and all pseudolites.

Figure 12: GPS Disciplined Oscillator

6.2.3 Short Range Acquisition

The third test consisted of three transmitters and one receiver used to conduct a short-range (<10m) acquisition test. Each pseudolite (Fig. 9), can act either as a receiver and transmitter. Seen in figure 13 is the test setup. Each transmitter sends out a 2.4GHz carrier wave modulated by a unique PRN code (as explained in Final Design section). The receiver, placed around 10m away, records raw data and runs it through the developed code in order to identify received signals, determine code phase and verify acquisition functionality.

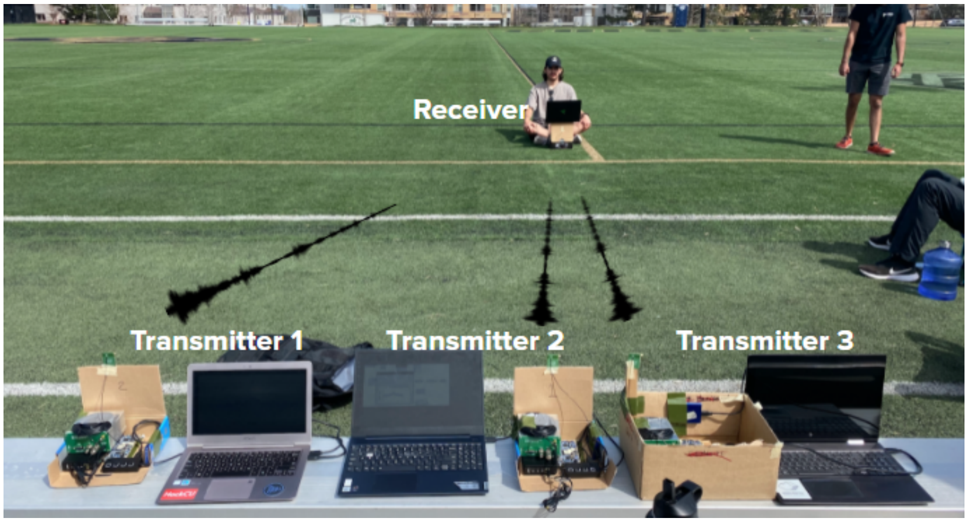

Figure 13: Short Range Acquisition Test Setup

6.2.4 Long Range Acquisition and Positioning

Our fourth and final test was identical to thee Short Range Acquisition Test, the only difference being larger distances were used (>100m). This test more closely resembles the goal of the project shown in our CONOPS (Fig. 2). A projection of this test can be seen below in figure 14.

Figure 14: Long Range Acquisition Test Setup

6.3 Results

6.3.1 Basic Communication Test

For the basic communication test, we were able to integrate most of the hardware and software in order to successfully send and receive an FM broadcast of Elton John's "Rocket Man." The transmit script involved reading .wav data from a file, modulating it onto an FM carrier at 2.4 GHz, transmitting it over the transmit antenna, receiving the signal using the receive antenna, demodulating the FM signal, and playing it over the computer speakers. While the sound played on the receiver end was somewhat noisy, we could easily tell that Rocket Man was playing.

This verified levels of success 1 and 2, which were to integrate the Ubuntu environment and all the dependencies for GNU Radio and LimeSDR. This also verified simple communication functionality for the overall system.

6.3.2 Oscillator Override

This test compared the GPS position solution performance of a LimeSDR + onboard oscillator and a LimeSDR + GPSDO. As seen in Figure 15, the use of a GPSDO greatly improved positioning performance by greatly reducing the spread of the calculated positions relative to the known truth. Quantitatively speaking, the 90% CEP radius (which contains 90% of all calculated solutions) was reduced from an initial value of 28.5m with the on-board oscillator, to 9m with the GPSDO. Additionally, mean positioning error was reduced from 39.5m to 16.5m with the GPSDO. This successfully verified both the integration of the GPSDO with a LimeSDR as well as improved performance due to the use a more stable oscillator.

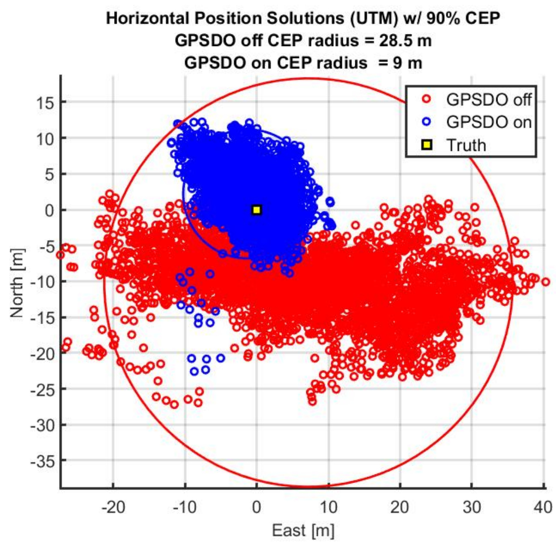

Figure 15: Clock Override Test

An additional bonus of successfully performing this test was that it demonstrated the functionality of the positioning algorithm (using GPS signals as opposed to P4LO pseudolite signals). Moreover, the 9m 90% CEP radius demonstrated this functionality within the 10m positioning accuracy set by the functional requirements.

6.3.3 Short Range Acquisition

For this test, three transmitters were setup to send out signals modulated by PRNs 1-3. On the receive side, the acquisition algorithm attempted to acquire PRNs 1-5. As shown in Figure 16, the algorithm succesfully acquired PRNs 1-3. Success in this case was defined as a specific PRN code having an acquisition metric higher than 2. The acquisition metric is the ratio between the 1st and 2nd strongest correlation peaks and is a measure of the signal-to-noise ratio of the received signal. A value of 2 for acquisition metric was chosen based on industry standards on what determines successful acquisition when working with GPS satellites.

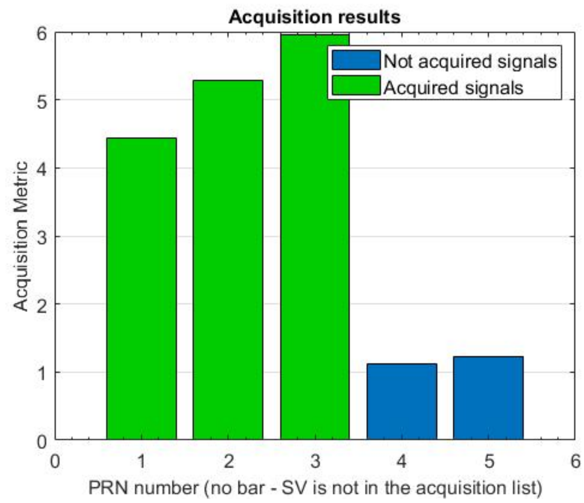

Figure 16: Short Range Acquisition Results

6.3.4 Long Range Acquisition and Positioning

For this test, acquisition was attempted with a setup very similar to that of the Short Range Acquisition Test, with the only difference being that the transmitters were placed at a distance of around 100m from the receiver. Although acquisition was successful, it came with two major caveats: longer signal processing times and the need for better antennas (on both the transmitters and receivers) as well as signal amplification (on the transmitter). The results shown in Figure 17 show that only a single signal (PRN 3) was successfully acquired. This is because the team was only able to acquire enough components to outfit a single transmitter with the necessary amplification hardware to improve transmission performance. The problem-solving approach the team took to get over this hurdle is outlined in the next section. Regardless of the problems faced on this test, it still verified that long-range acquisition was possible and set the stage for the verification of the overall positioning system.

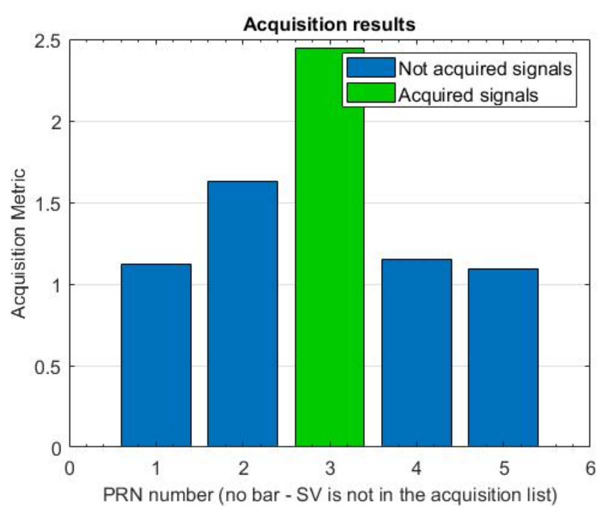

Figure 17: Long Range Acquisition Results

Long range acquisition debugging

The team tested a few key components of the hardware setup to determine why the setup couldn't acquire a signal at long range. The fact that the team was able to acquire the signal at all given a long integration time suggests that the signal was present, but at a low power. This means the transmitter wasn't transmitting enough power to be acquired in the presence of noise in the environment. The team conducted the following tests to isolate the issue:

1. Wired power test: this test is important to determine how much power was actually coming out of the LimeSDR transmit port at 2.4 GHz
2. Antenna test: after knowing how much power the LimeSDR is supplying, the team can test the antennas at a known separation and determine the loss due to the antennas
3. Upgraded system test: after adding an amplifier and using a different antenna, the team can test long range acquisition again

Wired Power Test

The first of these tests was the wired power test. Instead of connecting the SDR transmit port to an antenna, the team connected it directly to an oscilloscope that could measure signals at 2.4 GHz:

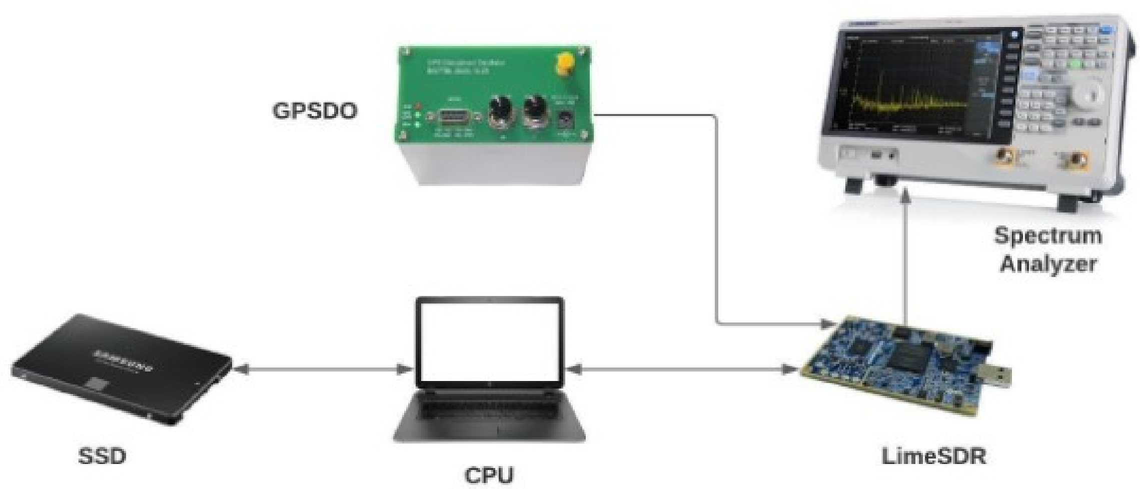

Figure 18: Wired Power Test Configuration

The team determined that the LimeSDR could output 1 mW of power (0 dBm) at 2.4 GHz, which, at 100% efficiency, leads to more received power at a distance of 100 m than the power received from GPS satellites 20,000 km away. The spread-spectrum CDMA modulation scheme should theoretically be able to operate on this power.

Antenna Orientation Test

The kind of patch antenna the team selected has a lot of applications, but in the case of mobile positioning systems where the antenna can change orientation frequently, a linearly polarized antenna is not a good selection. Linear polarization can work without loss if the transmit and receive antennas are perfectly aligned, but if the axis of polarization is misaligned, the power received could potentially drop to zero due to linear polarization losses.

Another potential source of loss is the wired connection between the antenna and the LimeSDR. The team noticed the connections were flush and tight at first, but as the antennas were disconnected and reconnected repeatedly, the connection became looser, with visible deformation on the antenna side of the UFL connection.

At a 1 meter separation between the patch antennas, with the orientation aligned as much as possible, the received power was -35 dBm.

Upgraded System Test

Finally, the team acquired a set of larger, more directional dipole 2.4 GHz antennas as well as a low noise amplifier. The new setup for the transmitter is shown below:

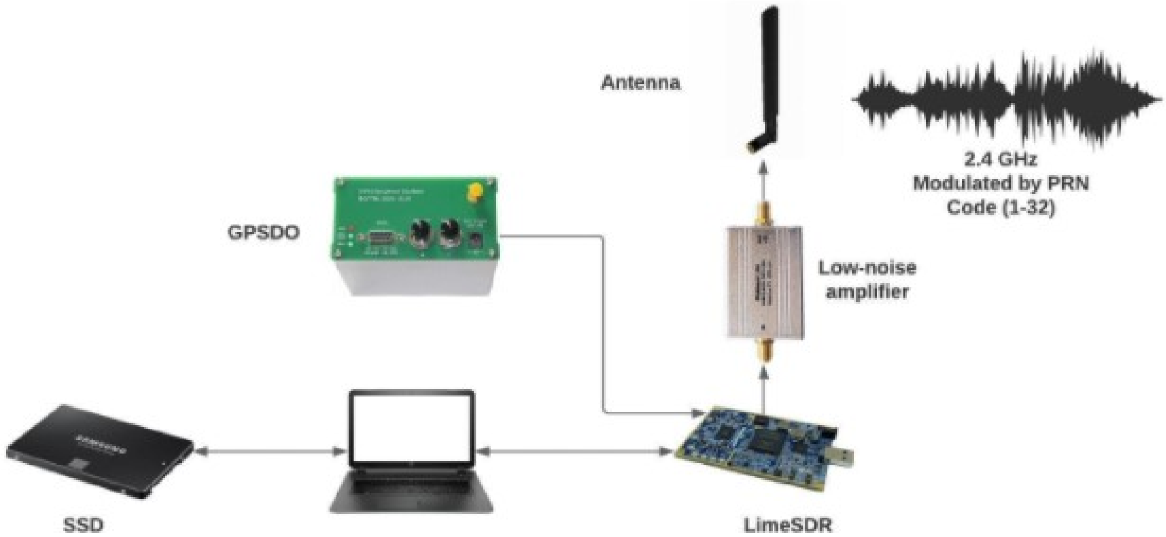

Figure 19: Upgraded System Transmitter Configuration

The receiver antenna was connected directly to the same 2.4 GHz oscilloscope, and the team determined at the same 1 meter separation as the other test that the received power was -15 dBm, or 100 times more power received than the original configuration.

With these changes, the team is confident that the hardware infrastructure is in place to conduct a successful long range acquisition test with three sets of the amplifier-antenna front end. Combined with the working positioning algorithm verified during the oscillator override test, this puts all the pieces in the place for a working positioning system.

7 Risk Assessment and Mitigation

Author: Ventura Morales

Throughout the project development, the team identified possible risks that could set back the project schedule, increase the budget, or even impede us in delivering on our critical requirements. These risks are quantified with the probability of occurrence and the severity of the risk itself. With this in mind we can designate an overall risk score between 1-5 based on likelihood and consequence. The risks include the near-far problem, where close-strong signals overpower weak-far signals; time synchronization between our pseudolites and our GPS receivers; team inexperience handling new software, such as GNU radio; signal interference in our testing environment; and electronic integration between our different hardware components. A more detailed definition of these risks can be seen in the risk log in figure 20 as well as their associated likelihood and consequence.

Risk ID	Description	Likelihood	Consequence
COM-1	Multiple User Communicating	5	2
COM-2	Signal Interference in testing environment	4	3
COM-3	Near-Far problem: Difficulty to hear weak-far signals from strong-close signal sources	4	5
SOF-1	Team inexperience handling new software. For example, GNU radio	3	4
SOF-2	Lime-SDR, Raspberry Pi, and other hardware components not communicating correctly	2	5
ELE-1	Time synchronization between pseudolite and GPS receiver	3	5
LOG-1	Hardware/Electronics don't arrive on time which sets back schedule of overall project	2	4
ANT-1	Antenna capabilities limitation, may not work on the stated performance values given on data sheet	4	4

Figure 20: Risk Control Log

Risks are placed in a risk matrix based on their likelihood and consequence (figure 21). Here we can see these risks are ranked between critical risk and low risk.

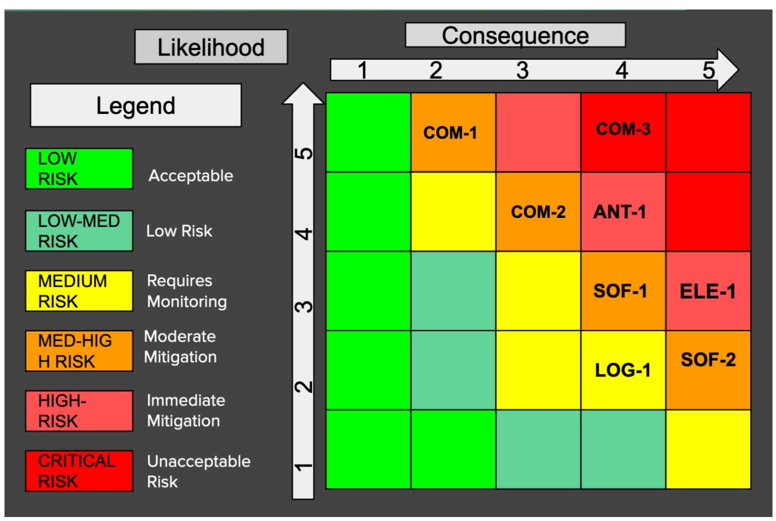

Figure 21: Risk Matrix

Our largest predicted risks were the near-far problem and the antenna's performance capability, followed by time synchronization, hardware setup, signal interference, and the communication link. Due to the likelihood and consequence of these risks, a mitigation plan was created to control these risks. The plan is shown below in figure 22.

Here we can see our mitigation approaches to compensate for these risks. To handle the near-far problem, the team leaned on our pseudolite geometry for our testing setup, only testing position and communication within our specified boundaries. While this is sufficient for our testing setup, future measures could include automatic gain control on the receiver end to compensate for overpowering signals. Because of our inexperience in handling new software and integration with our electronic components, we began early development on our SDR environment during the previous semester to ensure enough margin for testing. While signal interference is largely out of

Risk ID	Description	Mitigation Approach
COM-1	Multiple User Communicating	Design an optimal CDMA code for each user
COM-2	Signal Interference in testing environment	Change testing area to back up location (Platteville)
COM-3	Near-Far problem: Difficulty to hear weak-far signals from strong-close signal sources	• Test within pseudolite geometry boundaries • Automatic gain control
SOF-1	Team inexperience handling new software. For example, GNU radio	• Fall semester early development • Online tutorials, device users manual • External Resources: Dr Akos, Dr Rainville, Steve Taylor, Sam Holt
SOF-2	Lime-SDR, Raspberry Pi, and other hardware components not communicating correctly	• Fall semester early development. • Refer to overall block diagram, • Dive into specific functional block diagrams and debug issues (divide and conquer).
ELE-1	Time synchronization between pseudolite and GPS receiver	Conduct an initial test to verify limeSDR clock override with GPS-DO
LOG-1	Hardware/Electronics don't arrive on time.	Change provider of electronics to avoid schedule variance (Amazon, SparkFun, BestBuy,etc)
HDR-1	Antenna Limitations	Wired antenna test or borrowing faculty antennas to conduct a transmission and reception comparison test.

Figure 22: Mitigation Plan

our control, we conducted multiple tests to be aware of the likely interference and how it might affect signal acquisition. To mitigate the risk of antenna performance, the team tested the pseudolite system with different faculty provided antennas; transmitting and receiving signals at different antenna orientations. These tests allowed the team to better characterize the performance of these new antennas.

The team determined that after conducting these mitigation plans, risks will be lowered into a low category, where the risk is mitigated, or a medium category, where close monitoring is done to evade risk escalation. The post mitigation matrix is shown in figure 23.

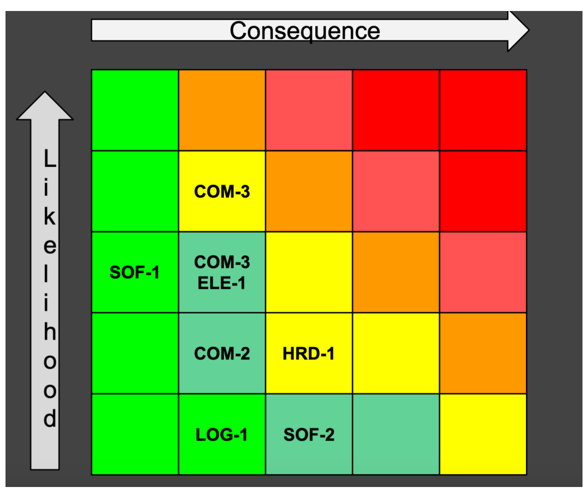

Figure 23: Post Mitigation Risk Matrix

Throughout the testing process, the team experienced some of the risks identified above, as well as ones that were not accounted for. The team experienced some signal interference in the testing environment, but it was not to an overwhelming extent. In addition, the GPSDO's arrived about 2 weeks late, ultimately posing a greater risk to the project then originally predicted. Antenna limitations proved the most troublesome of all of the risks that we predicted, with limited power and linear polarization causing issues with long range acquisition and preventing full scale tests. Without the antenna issues, the team would have been able to complete full scale testing and validated the entire system. Thus the team recommends that the customer instead utilizes a higher quality antenna, especially one that is circularly polarized. While the team predicted both high likelihood and consequence for this risk, it still exceeded the risk expectations.

The remainder of the risks that were estimated did not impact the project, as the team used proper risk mitigation techniques to prevent them from affecting results.

8 Project Planning

Author: Sam D'Souza

Below shows team P4LO's project plan in implementing our design solution. This plan is divided into four main sections: the organizational chart (highlighting our technical leads in each of our subsystems and their given sub-teams), the work breakdown structure (emphasizing the most important tasks in the completion of this project), the work plan (underlining key deliverables the team met and achieved in the spring semester), the cost plan (a comprehensive overview of the budget of our project and where the funding for this project was allocated) and finally the test plan (an overview of the scheduling and location of system tests).

8.1 Organizational Chart

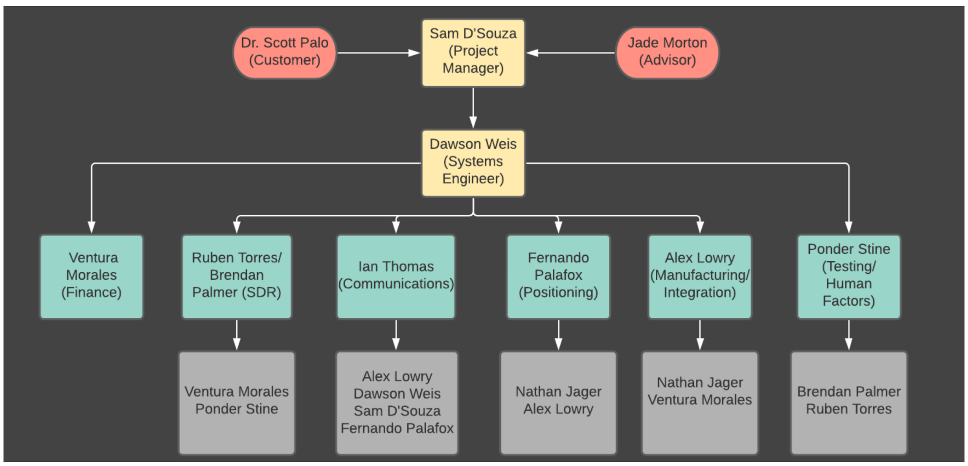

Figure 24: Team P4LO's Organizational Chart

Above in figure 24, we can see the organizational chart for team P4LO. Here, our team is working towards completing Dr. Scott Palo's (our customer) product request under the guidance of faculty advisor Dr. Jade Morton. We have broken our project into eight main components: project management, systems engineering, software defined radios (SDR), communications, positioning, manufacturing and integration, testing and human factors, and finance. Each team member is responsible for the management of each sub-component with the help of their respective sub-team.

8.2 Work Breakdown Structure

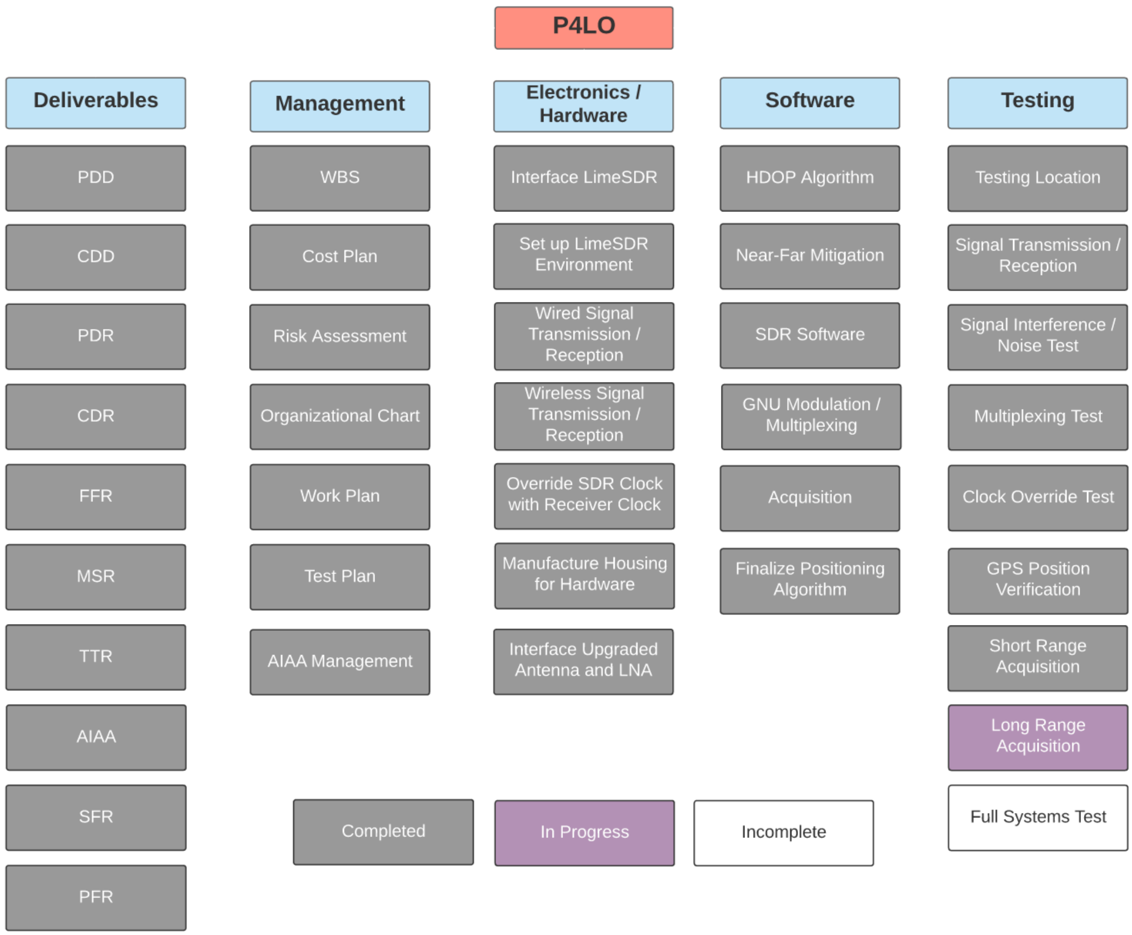

Figure 25: Team P4LO's Work Breakdown Structure

Above in figure 25, we can see team P4LO's work breakdown structure for the key work products in the project. The work shown has been divided into five main sections: class deliverables, project management, electronics and hardware, software, and testing. Class deliverables were derived from syllabus requirements and management tasks corresponded to scheduled deliverables and continuous project management.

For our electronics and hardware, the team had successfully accomplished setting up the computing environment to run our software defined radios and their corresponding software packages. The next step was to ensure our hardware was capable of meeting our requirements. To ensure that our LimeSDR had a stable clock, we interfaced it with our GPS-Disciplined-Oscillator, allowing us to utilize GPS signals to create oscillator stability for our project. From here the team then manufactured the housing for our system. While this housing was for function rather than aesthetics, the team needed a way to quickly setup pseudolites and conduct rapid testing, and thus the manufacturing team needed to construct a housing that met those requirements. Finally, the last work product within our electronics and hardware, was to interface our upgraded antenna and low noise amplifier to our system. This upgrade was needed to validate the inconsistencies within our original antenna selection and conduct additional tests to ensure system validity.

For our testing, the team started the spring semester having selected the testing location, demonstrated successful transmission and reception of signals and measured signal interference at the testing location. From here, the team then attempted to test the overridden clock of the system. The team conducted this testing by using our pseudolites as GPS receivers, and measured the difference in position solutions when compared to using the new oscillator and not using the new oscillator. The completion of this test verified that the interfacing process was a success as well as demonstrated the functionality of our positioning algorithm in comparison to the system functional requirements. The team then conducted close range acquisition, demonstrating that our system could successfully identify signals coming from multiple transmitters all transmitting within the same frequency. Once these close range tests were completed, the team then attempted to test long range acquisition where hardware inconsistencies limited the completion of the task. Once long range acquisition was complete the team would have conducted a full systems test in which our project would have demonstrated all the functional requirements placed upon it during the project inception.

8.3 Work Plan

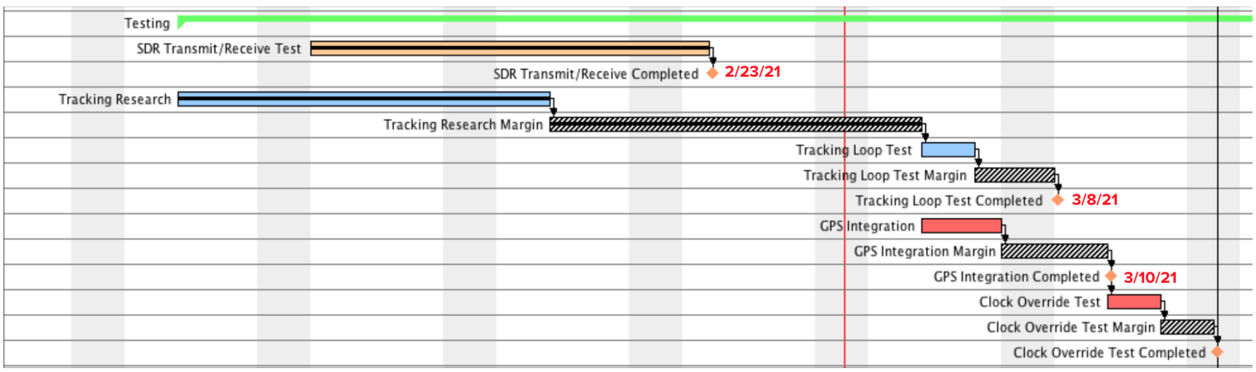

Figure 26: Team P4LO's Work Plan, Spring Semester

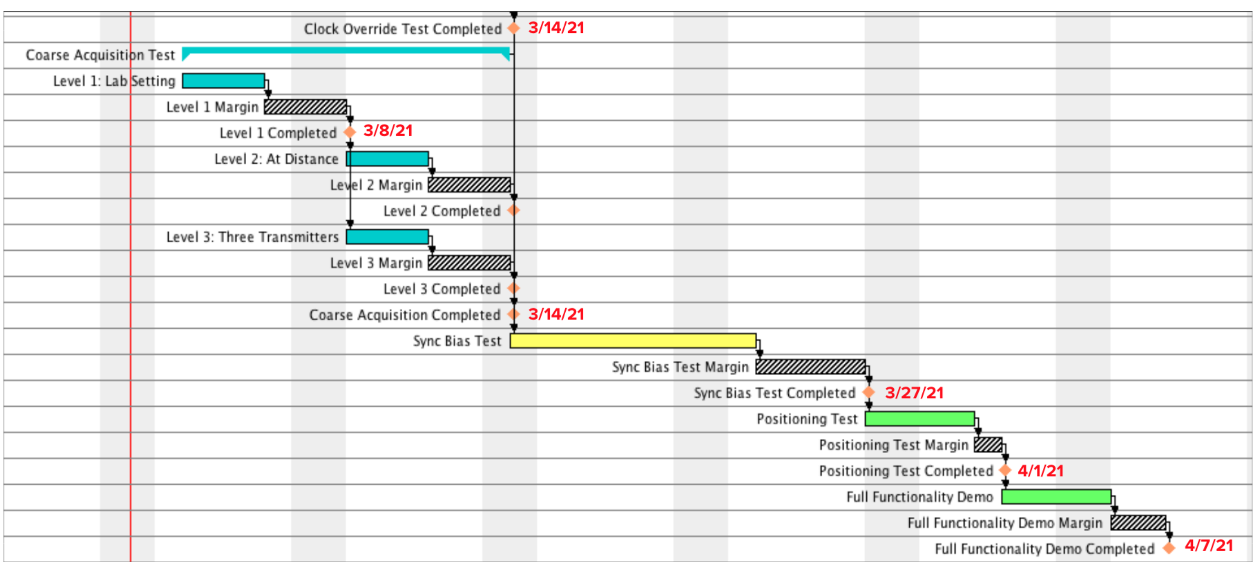

Figure 27: Team P4LO's Work Plan, Spring Semester 2

Shown above in figures 26 and 27 was team P4LO's work plan for the spring semester. Here we can see some of the main tasks for the spring semester, starting with our SDR transmit and receive testing which was completed on February 23rd. This task was a predecessor of the many of the following tasks and was thus the first part of the critical path of the project for the spring semester. Simultaneously, members of the communication sub-team were conducting tracking research to be able to implement and integrate tracking software given to us by our project advisor. Significant margin was built into this task as can be seen above as the team knew there would be a steep learning curve in understanding this material, and thus a significant margin was built into the task. In order to begin our full system testing, the team had to first get a stable oscillator within our SDR and thus had to integrate our GPS-Disciplined-Oscillator (GPSDO), with our Lime-SDR. As this task was critical in the continuation of the project, it was started in early March and allocated over two weeks of margin to ensure completion criteria of March 14th. Simultaneously, to maximize the bandwidth of our team, team P4LO was conducting a three staged coarse acquisition test. These tests included testing acquisition in a lab setting at a close distance, testing acquisition at a distance with a single transmitter, and testing acquisition at a distance with multiple transmitters. A leveled approach was used with a week of margin for each test in an attempt to complete the leveled testing by March 14th. From here, the team then planned on to conduct a synchronization bias test, a test in which would allow us to synchronize all three transmitters such that we can accurately compute position. As this test involved a significant amount of testing, one week of margin was allocated to ensure for the completion of this task. Once the synchronization was complete, the next step in the critical path was to conduct positioning tests, and verify these tests with respect to our requirements. These tests were projected to be completed by April 1st, allowing us one week of margin to document and record a final demo presentation for the projects conclusion. This demo was projected to be completed by April 7th, giving the team significant margin to complete the final presentations and class deliverables before the projects end date in late April. While adequate margin was associated for the majority of the work breakdown structure products, unfortunately there were unforeseen hardware difficulties encountered in coarse acquisition testing. The team successfully created and implement the software necessary to acquire multiple signals at a distance, however antenna polarization inconsistencies didn't allow for the team to complete these tests in a consistent enough manner to continue with the critical path. This in turn forced a roadblock in which didn't allow for the team to meet the final work breakdown structure products. As highlighted in earlier sections however, the team was creative in demonstrating capability for the final work products such that if this roadblock had been removed, the team would have been on track to hit all key work products within their respective completion dates.

8.4 Cost Plan

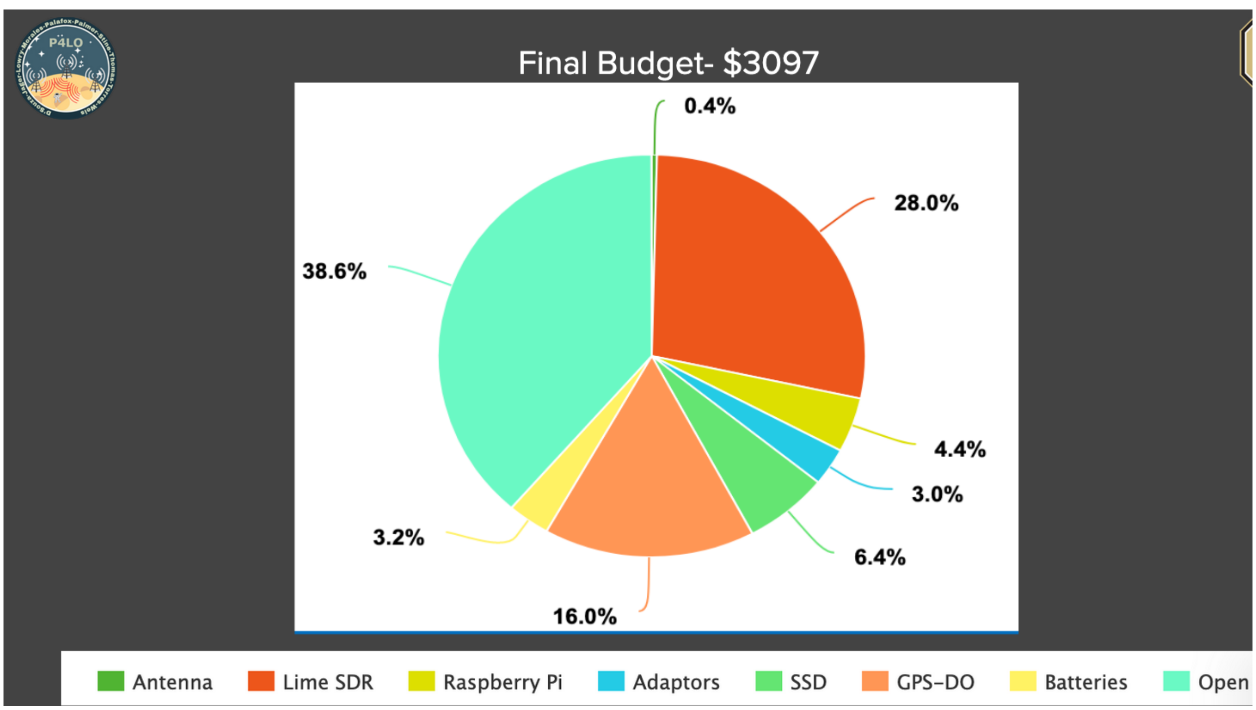

Figure 28: Team P4LO's Budget

Project P4LO had an allowable budget max of $5000 USD dollars. Shown in figure 28 we can see the teams budget pie chart. The two major components of the teams budget were the purchasing of GPS Disciplined Oscillators and

Lime SDR's. Next we have the SSD's, Raspberry pi's, mobile power pack batteries and miscellaneous adaptors as well as the antennas. All of these components were used to develop the pseudolites. The final purchasing order total accumulated to be $3097 dollars which is equivalent to using only 61.94% of the allowable budget. A more detailed overview of our overall budget can be seen in figure 29.

			Team P4LO Budget	
Project Budget	**$5,000.00**			
			Communications Hardware	
Item	Individual Cost	# of Items	Subtotal Per Set	Supplier
Antenna	$4.95	4	$19.80	SparkFun
Lime SDR	$349.95	4	$1,399.80	SparkFun
Raspberry Pi	$55.00	4	$220.00	SparkFun
U.fl to BNC Adaptor	$7.67	4	$30.68	Amazon
Power Pack	$39.99	4	$159.96	Amazon
Sabrent 4-Port	$10.99	4	$43.96	Amazon
Testing Tripods (5 ft)	$27.99	1	$27.99	Amazon
			$0.00	
		Storage Hardware / Final Clocks / Post-CDR Components		
Item	Individual Cost	# of Items	Subtotal Per Set	Supplier
External SSD Card (500 Gb)	$79.99	4	$319.96	Amazon
GPS Reciever Clock	$200.00	4	$800.00	Amazon
Best Buy USB Extenders	$14.99	5	$74.95	
			Budget Balance	
Cumulitive Order Total	**$3,097.10**	Used Budget Percentage	**61.94%**	
Available Budget Balance	**$1,902.90**	Available Budget Percentage	**38.06%**	

Figure 29: Team P4LO's Detailed Budget

8.5 Test Plan

Test Plan

Testing Procedure	Date Scheduled	Location
Signal Reception Test	February 23rd	Engineering Center
Modulation and Multiplexing Test	March 1th	Engineering Center
Short Range Acquisition	March 8th	Engineering Center
GPS Override	March 14th	Foothills Park
Long Range Acquisition	March 25th	Foothills Park
Upgraded Pseudolite	April 1st	Engineering Center

Figure 30: Team P4LO's Test Plan

Shown above in figure 30, is team P4LO's test plan. The two main testing locations in which tests were conducted were the Engineering Center, for close range tests, and Foothills park, our final testing location that allowed our team to test at distances closer to 200 meters. These tests were scheduled in a manner that built upon the previous and the completion of all tests will validate the functional requirements set about by our customer.

9 Lessons Learned

Some of the biggest lessons learned were in risk reduction for our customer Dr. Scott Palo, and ultimately the LunaNet program as a whole. The importance of a thorough link budget analysis that includes an in-depth performance analysis of the selected antennas, radios, and amplifiers was learned the hard way as the performance of the teams antenna did not meet expectations set by the hardware documentation. The issues with signal acquisition can be mostly attributed to the choice in antenna used for the project. With a better antenna it is very likely that the team would have been able to have improved transmit power and therefore improved reliability in acquisition to move onto positioning tests. Unfortunately, by the time this discovery was made it was too late in the semester to acquire new hardware to test all three transmitters, even though the team was still able to verify the positioning algorithm via GPS. Additionally, proper radio selection was another lesson learned. Choosing a radio that could more easily interface with an external clock like the GPSDO would have cut down on time figuring out the interfacing of the hardware.

Early in the project it was anticipated by the team as well as the PAB that software packages like GNU radio would be a major challenge, as it has been for other teams in the past. These software packages ended up being one of the elements that was completed early on, largely without issue. In hindsight, the hardware was more complicated and had a greater risk than the software for the project.

Another significant lesson learned was all the experience the team gained from working hands-on with the hardware through the learning curve and troubleshooting process. Before this project, none of the team members had any experience with software defined radios (SDRs) and only a couple had any idea how positioning systems worked. However, by the end of the project the team had overcome a multitude of hardware-software interfacing difficulties in order to get working pseudolites which could connect with an external oscillator, transmit and receive specific navigation signals, and even compute position within 10m accuracy using GPS.

Due to the number of challenges the team had to overcome, an important theme throughout the year was to stay flexible within our design space. For example, when the LimeSDR could not easily synchronize with the GPSDO without complicated FPGA programming, we had to adjust and create a new method for synchronizing the clocks on the transmitter pseudolites to make accurate positioning possible. In the future, we would recommend a different radio be used for a positioning system, however, in case there is no other option, a methodology was developed that should allow the LimeSDRs to work together in the future.

And finally, field testing had some growing pains that were able to to be overcome to improve overall testing efficiency. While initial tests took many hours to set up and coordinate, the team got more organized with pseudolite housings and checklists to make testing run much smoother. A big improvement in testing efficiency was using SSDs and laptops at each of the pseudolite transmitters instead of our initial plan of using Raspberry Pis. One of the most significant improvements in test efficiency was being able to run the receiver through Matlab instead of GNU radio. This removed the need to transfer data from the Ubuntu run GNU radio to the Windows run Matlab script for analyzing data. Tests could then be run and analyzed with a few keystrokes and only seconds of waiting time instead of spending minutes collecting data, transferring files, and running an analysis script. This greatly increased the speed at which we were able to run various tests and know the results of each test.

Overall, the project was a great success for team P4LO. Advice for upcoming seniors would include ensuring they conduct a thorough and inclusive look at all possibilities of risk in their project early on. Ensuring they are thinking about all risky endeavours early in the project would allow them to plan tests accordingly that ensures that these risks are mitigated early on. Our team ensured aspects of our hardware would meet the manufacturers requirements and unfortunately did not test their performance early on. Future seniors will have a much improved experience if they implement this advice.

10 Individual Report Contributions

- Ruben Hinojosa Torres: In charge of the GNU Radio and LimeSDR setup. GNU Radio portion of software trade study. SDR software package detailed design.
- Sam D'Souza: In charge of project and team management. Organizational Chart, Work Breakdown Structure, Work Plan, Cost Plan, Functional Requirements.
- Fernando Palafox: Positioning lead. In charge of positioning system analysis and development. Positioning system trade studies, baseline design and detailed design.
- Ian Thomas: Communications lead: communications design options (content and writing), trade studies (content and writing), baseline design (content and writing), detailed design (content and writing).
- Nathan Jager: Positioning and Manufacturing Colead: Conducted trade study and research between one way and two way ranging, research on Lunar gps systems, and wrote CONOPS.
- Dawson Weis: Systems Engineer: In charge of higher level overview of all aspects. Requirements Development, Project Purpose, Functional Block Diagrams, Functional Requirements
- Alex Lowry: Manufacturing/Integration Lead: Suported Communications and Positioning systems. Project Overview, Levels of Success, Requirements and their Development, and full report proof read and editing.
- Ventura Morales: Conducted antenna trade-studies, baseline and detailed design of the antenna, risk analysis and mitigation. Financial budget for next semester was done.
- Brendan Palmer: SDR Colead and Testing: Working with SDR software packages and trade studies(content and writing). Verification and Validation(Content and writing). Baseline design(content). Background Research(content)
- Ponder Stine: Human factors/Testing lead: background research (content), baseline design (content), levels of success table (content and writing), test plan (content and writing), verification and validation (content and writing)

References

[1] LunaNet: Establishing a Flexible and Extensible Lunar Exploration Communication and Navigation Infrastructure : NASA Goddard Space Flight Center.

[2] D. J. Israel et al., "LunaNet: a Flexible and Extensible Lunar Exploration Communications and Navigation Infrastructure," 2020 IEEE Aerospace Conference, Big Sky, MT, USA, 2020, pp. 1-14. doi: 10.1109/AERO47225.2020.9172509 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=arnumber=9172509isnumber=9172248

[3] Jerry Colen, "NASA Selects Universities for Collaborative Development of Small Spacecraft Technologies." NASA Ames Features, Published on March 11th, 2020, https://www.nasa.gov/feature/ames/nasa-selects-universities-for-collaborative-development

[4] Brian Dunbar, "Moon's South Pole in NASA's Landing Sites" National Aeronautics and Space Administration, April 19, 2019, https://www.nasa.gov/feature/moon-s-south-pole-in-nasa-s-landing-sites

[5] Richard B. Langley (May 1999). "Dilution of Precision" (PDF). GPS World.

[6] Wolff, Christian. "Radar Accuracy." Radartutorials.eu, Christian Wolff, 2018, www.radartutorial.eu/01.basics/Radars%20Accuracy.en.html.

[7] "Tutorials." Tutorials - GNU Radio, 9 Sept. 2020, wiki.gnuradio.org/index.php/Tutorials.

[8] "Guided Tutorial Introduction." Guided Tutorial Introduction - GNU Radio, 2020, wiki.gnuradio.org/index.php/Guided_Tutorial_Introduction.

[9] "Guided Tutorial PSK Demodulation." Guided Tutorial PSK Demodulation - GNU Radio, 2020, wiki.gnuradio.org/index.php/Guided_Tutorial_PSK_Demodulation.

[10] "Gr-Limesdr Plugin for GNURadio." Myriad, Myriad RF, wiki.myriadrf.org/Gr-limesdr_Plugin_for_GNURadio.

[11] Noordhof, Jan. "How Does Modulation Work?" Tait Radio Academy, Tait Communications, 2020, www.taitradioacademy.com/topic/how-does-modulation-work-1-1/.

[12] Hsu, Eric. "Modulation Schemes for Satellite Communications." Keysight Blogs, Keysight Technologies, 2020, blogs.keysight.com/blogs/tech/rfmw.entry.html/2020/08/24/modulation_schemesf-6MpZ.html.

[13] NI. "Pulse-Shape Filtering in Communications Systems." NI, NI, 2020, www.ni.com/en-us/innovations/white-papers/06/pulse-shape-filtering-in-communications-systems.html.

[14] eValidate. "Binary Phase Shift Keying (BPSK) Modulation." EValidate, EValidate, www.evalidate.in/lab2/pages/BPSK-mod/BPSK/BPSK_I.html.

[15] Bharati, Subrato, et al. "Implementation of ASK, FSK and PSK with BER vs. SNR Comparison over AWGN Channel ." Arxiv.org, Cornell University, 2020, Implementation of ASK, FSK and PSK with BER vs. SNR comparison over AWGN channel .

[16] Editor, Author: Technical. "Advantages and Disadvantages and Applications of Time Division Multiplexing (TDM)." Polytechnic Hub, Polytechnic Hub, 16 Nov. 2017, www.polytechnichub.com/advantages-disadvantages-applications-time-division-multiplexing-tdm/.

[17] "OFDM vs CDMA | Difference between OFDM and CDMA." RF Wireless World, RF Wireless World, 2012, www.rfwireless-world.com/Articles/OFDM-vs-CDMA.html.

[18] Rashed, Ahmed Nabih, and Mohammed Salah Tabbour. "The Trade Off Between Different Modulation Schemes for Maximum Long Reach High Data Transmission Capacity Optical Orthogonal Frequency Division Multiplexing (OOFDM)." Researchgate, Researchgate, May 2018, The Trade Off Between Different Modulation Schemes for Maximum Long Reach High Data Transmission Capacity Optical Orthogonal Frequency Division Multiplexing (OOFDM).

[19] Laaraiedh, Mohamed, et al. “Cramer–Rao Lower Bounds for Nonhybrid and Hybrid Localisation Techniques in Wireless Networks.” Researchgate, EUROPEAN TRANSACTIONS ON TELECOMMUNICATIONS, 2011, www.researchgate.net/publication/235936422_Cramer–Rao_lower_bounds_for_nonhybrid_and_hybrid_localisation_techniques_in_wireless_networks/link/5bfbba00a6fdcc53881ad9

[20] Alexandru Csete OZ9AEC. Welcome to Gqrx, gqrx.dk.

[21] Alexandru Csete OZ9AEC. “Supported Devices.” Qqrx SDR, gqrx.dk/supported-hardware.

[22] Alexandru Csete OZ9AEC. “Gqrx Design.” Gqrx SDR, gqrx.dk/doc/gqrx-design.

[23] MyriadRF. LimeSuiteGUI, wiki.myriadrf.org/LimeSuiteGUI.

[24] MyriadRF. Lime Suite. wiki.myriadrf.org/Lime_Suite.

[25] Myriad. “Getting Started with LimeSDR-USB and LimeSuiteGUI.” Myriad, wiki.myriadrf.org/Getting_Started_with_LimeSDR-USB_and_LimeSuiteGUI.

[26] Carey, Daniel, et al. “STATISTICAL MODELING OF BIT-ERROR-RATES IN ASYNCHRONOUS MULTICARRIER CDMA AND DIRECT-SEQUENCE CDMA SYSTEMS.” Eurasip.org, EURASIP, www.eurasip.org/Proceedings/Eusipco/Eusipco2004/defevent/papers/cr1525.pdf.

[27] Septentrio Satellite Navigation, "PolaRxS: Ultra Low Noise GNSS Receiver for Ionospheric Scintillation Monitoring", 2010

11 Appendix

11.1 Communications Design Options

Author: Ian Thomas

11.1.1 Modulation

Amplitude Shift Keying
Amplitude shift keying is a type of digital modulation technique where the carrier frequency's amplitude is changed according to which bit or set of bits is meant to be transmitted [12]. A figure detailing the process is shown below:

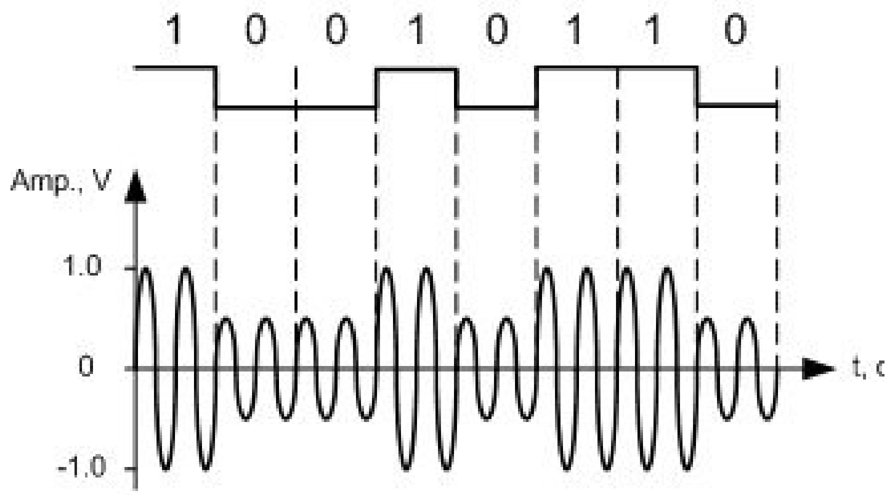

Figure 31: Binary Amplitude Shift Keying [14]

The figure shows a form of ASK called binary amplitude shift keying (BASK), in which two amplitudes are used to represent the digital bit states 0 and 1. This can be depicted in a *constellation diagram*, in which various amplitudes and phases used to modulate the signal are represented as points on the constellation diagram. For example, the constellation diagram for BASK is as follows:

In a constellation diagram, the axes are the *in phase* axis (I) and *quadrature* axis (Q). Their units are volts which represent signal amplitude (so the amplitude of an arbitrary point is its distance away from the origin), and the phase of a constellation point is described by the angle a line drawn between it and the origin makes with the $+I$ axis. A key feature of pure amplitude shift keying is that all the constellation points lie on the same ray from the origin (they all have the same phase). Another example of amplitude shift keying is 4ASK, shown below:

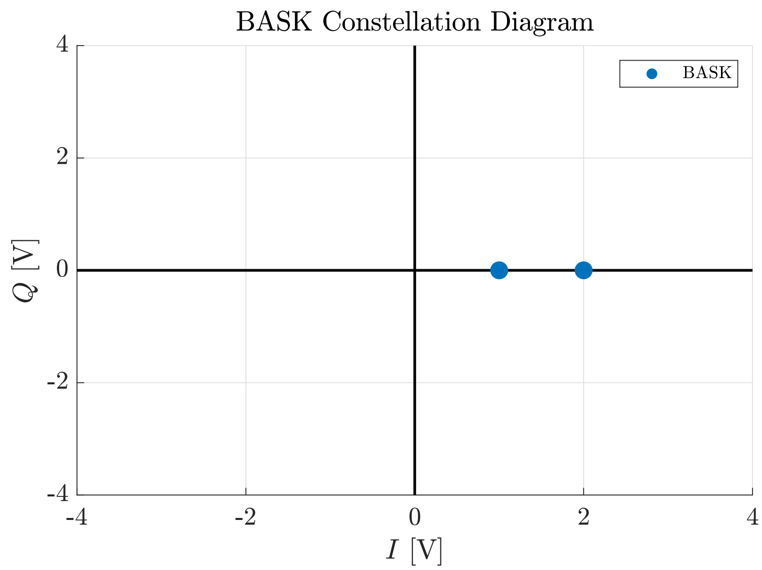

Figure 32: Constellation Diagram for Binary Amplitude Shift Keying

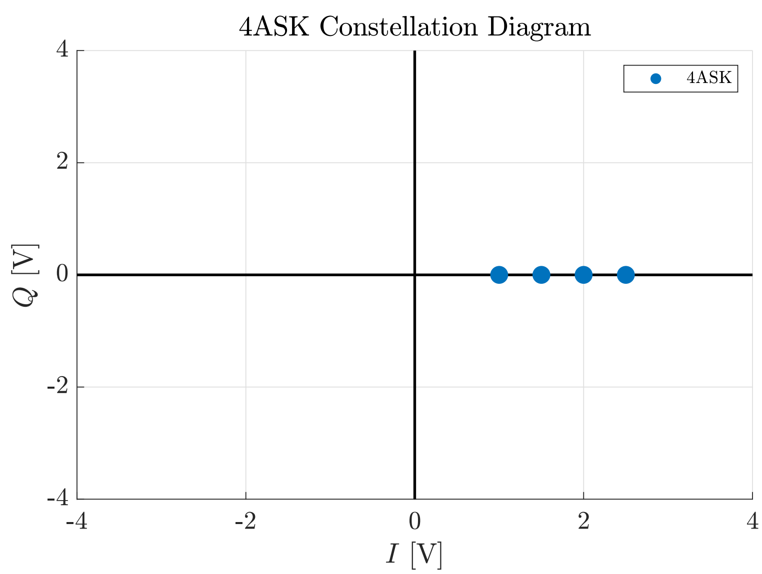

Figure 33: Constellation Diagram of 4ASK Signal

Here, each point in the signal would represent two bits:, either 00, 01, 10, or 11. The mapping of the bit sequences (symbols) to the constellation points is up to the designer of the signal. The benefits of pure ASK are as follows:

- ASK is simple to implement
- Scalable to allow for faster data rates
- ASK can be combined with other modulation schemes to make a more robust hybrid scheme

Some of the drawbacks of ASK are as follows:

- ASK does not utilize the constellation space effectively. The closer constellation points are to each other, the less energy it takes for noise or interference to cause a signal at one constellation point to be interpreted as one from another constellation point, so it is good to maximize the spacing of the constellation points in order to minimize bit error rate [12]

- ASK requires a good amplifier to work properly, or risk increasing bit error rate due to amplifier noise and amplitude/phase shifts due to amplifier nonlinearities (see section on Quadrature Amplitude Modulation) [12]
- ASK can lead to sharp transitions in the carrier signal, so in order to stay within a certain bandwidth, a pulse shaping filter must be applied to the signal to make the transitions smoother [13]

Frequency Shift Keying
Frequency shift keying, as opposed to amplitude shift keying, changes the frequency at which the carrier signal is transmitted in order to convey bits. A figure is shown below:

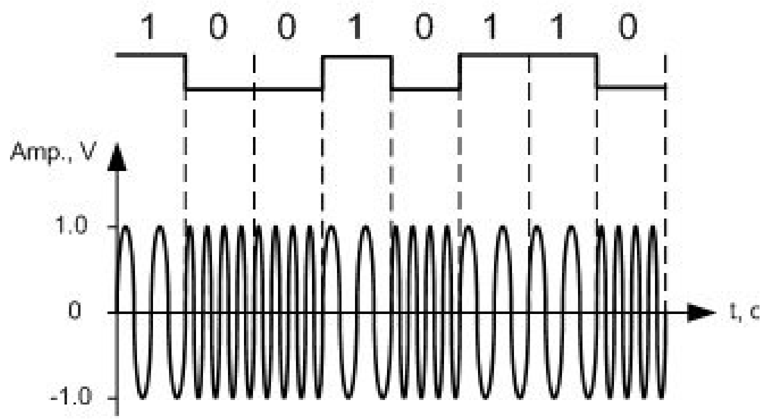

Figure 34: Frequency Shift Keying

Some benefits to FSK are listed below:

- FSK is simple to implement
- Many different frequencies can be utilized to increase data rate by sending multiple bits per symbol
- There aren't sharp transitions in an FSK signal, so there is no need to apply a pulse shaping filter to keep it within bandwidth

Some drawbacks to FSK are listed below:

- FSK must utilize many different frequencies, which removes some bandwidth to be possibly used for multiplexing signals. The data rate is heavily limited by bandwidth.
- The communication link is to be used between users on the surface on the moon and satellites in orbit around the moon. The velocity of the satellites is very large and variable with respect to the lunar surface users, so the Doppler shift on signals could significantly increase the bit error rate of the signals

Phase Shift Keying
Unlike ASK and FSK, phase shift keying uses the phase of a carrier signal to transmit individual bits. A diagram for binary phase shift keying, or BPSK, is shown below:

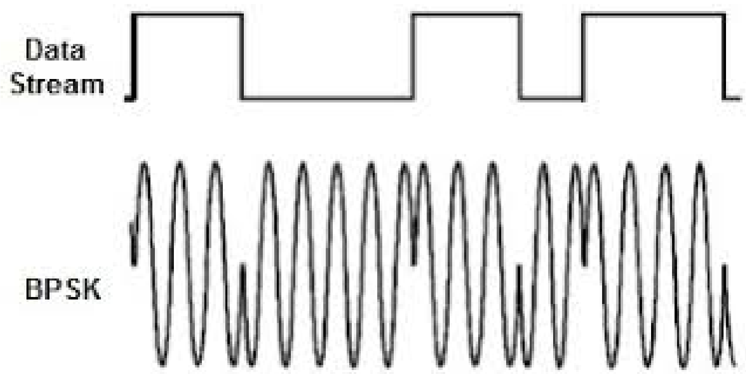

Figure 35: BPSK Modulation onto Carrier Signal [14]

BPSK uses two phases, in phase and exactly out of phase, to represent two states of a digital bit. The figure below shows the constellation diagram of BPSK:

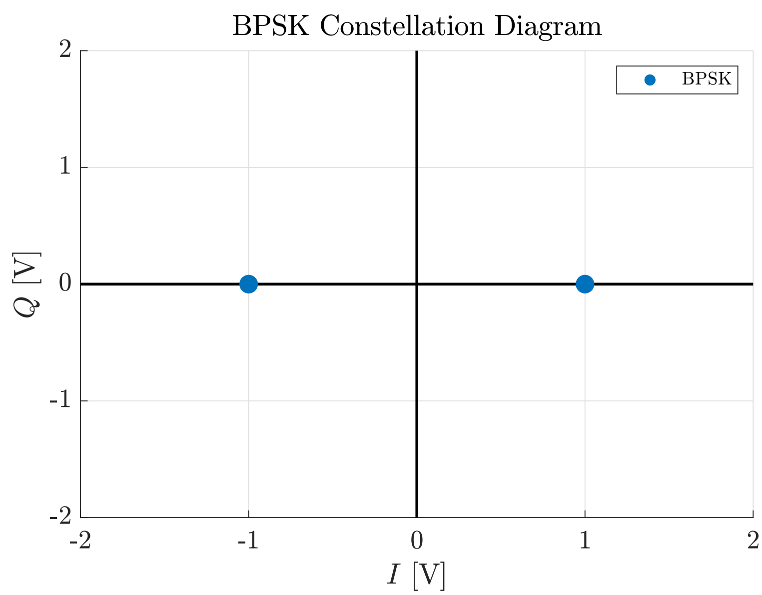

Figure 36: Constellation Diagram of BPSK signal

The constellation diagram above shows two points with the same amplitude, except 180°apart. This can be generalized into higher order PSK, such as quadrature phase shift keying (QPSK). The constellation diagram for QPSK is shown below:

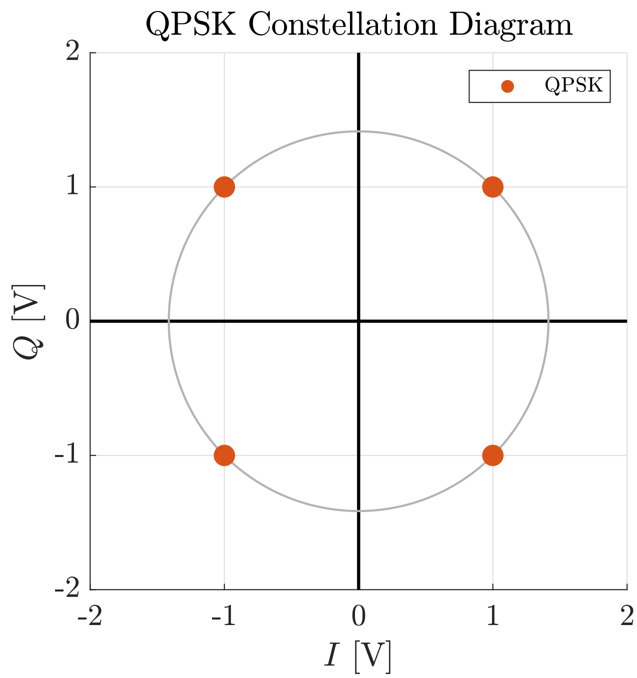

Figure 37: QPSK Constellation Diagram

All the constellation points lie on the same circle centered around the origin, and are separated in phase by 90°. Each constellation point represents a two bit symbol. Like the other main forms of digital modulation, it is possible to include more constellation points on this circle (more phases) to encode more bits per symbol. Here are a list of benefits of phase shift keying:

- A nonlinear amplifier will affect all the points on the constellation of a PSK in similar ways
- There is no bit error rate cost in moving from BPSK to QPSK (but there is bit error rate cost in moving to higher PSK schemes)
- The bit error rate is lower than that of FSK and ASK [15]

Here are some of the drawbacks of pure PSK:

- Constellation spacing could increase if combined with ASK [12]
- It is one frequency, so it is prone to interference at that frequency

Quadrature Amplitude Modulation

Quadrature amplitude modulation is a combination of phase and amplitude shift keying, where the minimum number of constellation points to make improvement over PSK is 16. The constellation diagram for 16QAM is as follows:

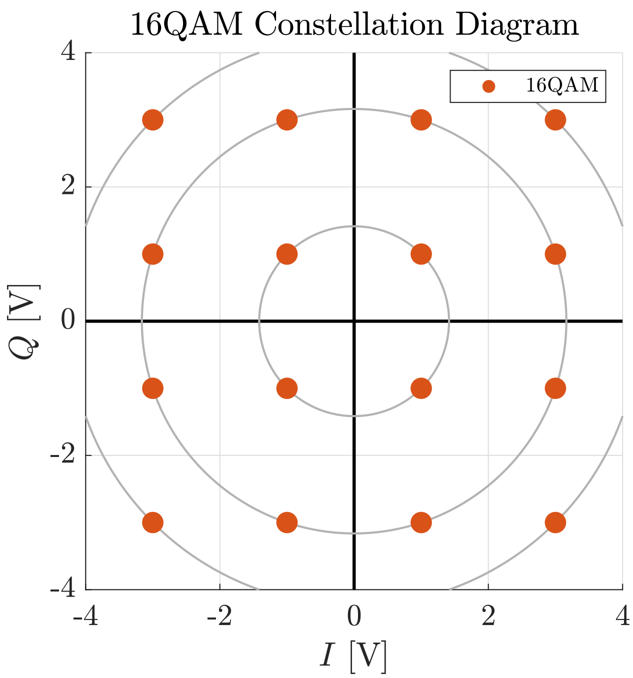

Figure 38: 16QAM Signal Constellation

The constellation diagram shows 16 constellation points spread out fairly evenly across 3 amplitudes and 12 different phases. It's certainly clear that the separation for the number of constellation points is better than what either pure ASK or pure PSK can do by themselves, but the cost of having this many constellation points will still be present in the bit error rate. Additionally, amplitude modulated signals need to use signal amplifiers to boost the signal to different levels for transmission. This amplification process can have nonlinearities which affect both the amplitude and phase of a 64QAM constellation point as a function of its amplitude, as shown in the figure below [12]:

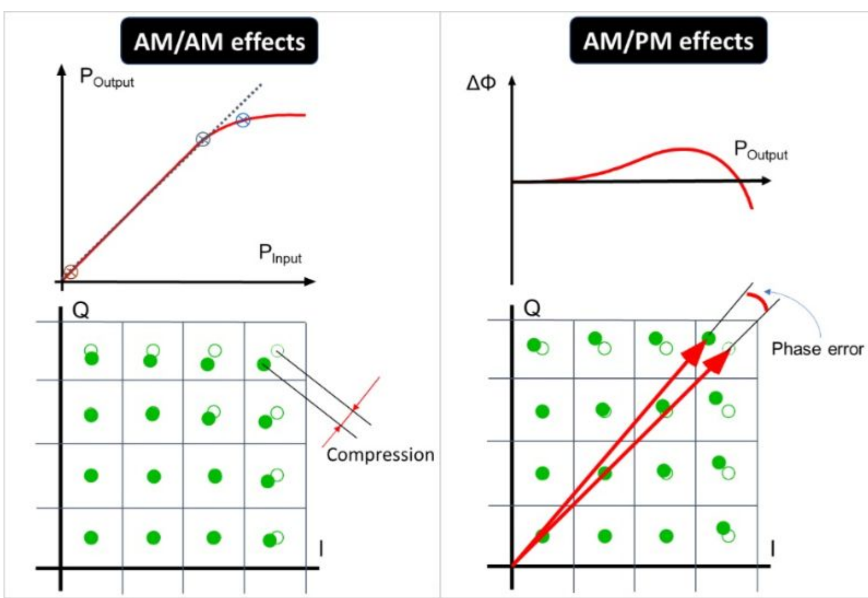

Figure 39: Amplitude and Phase Nonlinearities in 64QAM constellation plot [12]

The figures show the first quadrant of a 64QAM signal, but the compression and phase error is present in any amplitude-modulated signal, not to mention any additional noise caused by the amplifier. It is important to note many ground and space-based systems use QAM for modulation, and its primary benefit which is its sharp increase in data rate comes at the cost of a higher bit error rate.

11.1.2 Multiplexing

Time Division Multiplexing

Time division multiplexing is the simplest of multiplexing techniques, in which different terminals get certain time allocations to send their signals. This means only one signal is being sent through the medium at a time, but this could be optimized to maximize throughput. Some of the pros of time division multiplexing are:

- There aren't any additional multiplexing steps to take other than synchronization between the sender and receiver
- Each signal can make full use of the available bandwidth [16]

Some of the drawbacks include:

- Synchronization between the sender and the receiver to know which signal is transmitting at any given time is hard [16]
- Synchronization between multiple senders (in the case of GPS satellites) would require extremely accurate timing information between satellites and between satellites and the receiver

Frequency Division Multiplexing

Frequency division multiplexing is where many users are assigned their own frequency within the bandwidth of the carrier and then a receiver can use Fourier techniques to split up the joined signal into their frequency components and retrieve the original signal [17]. A variation of FDM, called orthogonal FDM or OFDM, is used widely in many telecommunications applications because of its high spectral efficiency (how well a signal uses its bandwidth) [18]. It achieves this by using several orthogonal closely spaced frequencies which can each carry their own signal. Some benefits of FDM (and its variants) include:

- Potential for very high spectral efficiency if closely spaced frequencies are used [17]
- High data rate and many users [18]
- Can be combined with other single-frequency multiplexing techniques to form a better solution
- Has a wide range of telecommunications heritage (OFDM is the primary multiplexing technology of 4G) [17]

Some of the drawbacks include:

- Closely spaced frequency bands are very sensitive to Doppler shift [18]
- Number of users is limited by number of frequencies available for use within bandwidth

Code Division Multiplexing

Code division multiple access (CDMA) is a technique in which a data signal is combined with a much faster data rate spreading code, which can then be modulated onto a carrier frequency to be received by a receiver with the same spreading code [17]. An image of the spreading process is shown below:

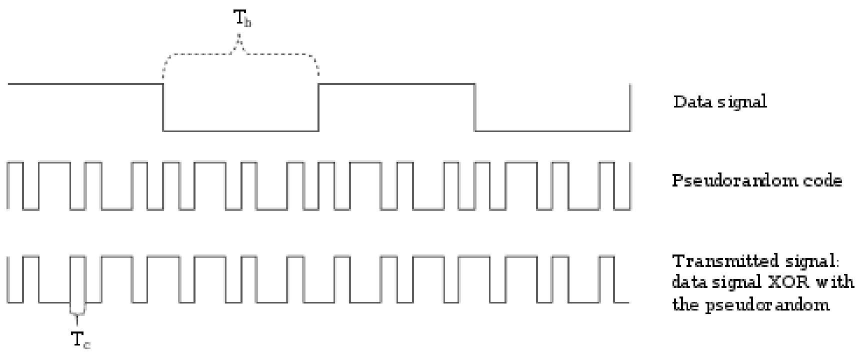

Figure 40: CDMA Spreading

The spreading codes have the property of being mutually orthogonal (or highly orthogonal) - which means their cross-correlations are perfectly (or nearly) zero for all time. This means that a receiver can pick up many different CDMA signals from many different transmitters and they will all look like random noise except the one which matches the code the receiver is using. CDMA is the technology currently in use for GPS because each GPS satellite can transmit using its own unique code, allowing for identification and separation of signals on the receiver end (the receiver can tell which satellite is sending each message). Some of the benefits of a CDMA system include:

- CDMA supports multiple users on the same carrier frequency, which reduces the bandwidth of the transmitted signal (even though the bandwidth of the digital signal prior to modulation increases) [17]
- If the spreading codes are good, CDMA signals have potentially very low inter-signal interference
- CDMA signals have a legacy of being used for navigation and GPS

Some drawbacks of CDMA include:

- The spreading codes only correlate if they are perfectly in phase, which means a receiver has to perform a phase correction in order to correctly demultiplex the signal
- The number of users on a CDMA system is limited by the ratio of the bit rate of the spreading code to data rate, called the spreading factor [17]

Within CDMA, there are multiple options regarding how to generate the spreading code for different bit error rate results. Additional consideration will have to be given to modulation with CDMA to determine if a dataless acquisition signal can be sent in addition to the data to increase the time of signal acquisition.

11.2 SDR Software Package Installation

Author: Ruben Hinojosa Torres

```
[1] - LimeSuite Installation: (PPA Installation)
Note: If this installation does not work, it might be necessary to have Ubuntu 18.04 or 18.10.
-> sudo add-apt-repository -y ppa:myriadrf/drivers
-> sudo apt-get update
-> sudo apt-get install limesuite liblimesuite-dev limesuite-udev limesuite-images
-> sudo apt-get install soapysdr-tools soapysdr-module-lms7
To open the LimeSuiteGUI simply type the command:
-> LimeSuiteGUI
To close, simply close the window.
If you have access to the LimeSDR, you can verify the GUI can talk to it by typing the command:
-> SoapySDRUtil --find="driver=lime"

[2] - GNU Radio Installation: (PPA Installation)
Note: The Release version must be 3.8.2.0
Note: If this installation does not work, it might be that Qt
     dependencies are missing and therefore will require more looking into.
-> sudo add-apt-repository ppa:gnuradio/gnuradio-releases
-> sudo apt-get update
-> sudo apt install gnuradio
To optimize GNU Radio kernel usage type:
-> volk_profile
To Open GNU Radio GUI:
-> gnuradio-companion
To close, simply close window.

[3] - Gr-limeSDR Plugin Installation: (Source Code Installation)
Note: This is the most problematic and therefore requires extra attention.
[3.1] - Installing Dependancies:
-> sudo apt-get install libboost-all-dev swig
Just in case:
-> sudo apt-get install gnuradio-dev
[3.2] - Building gr-limesdr from source: (Only for GNU Radio 3.8!)
-> git clone https://github.com/myriadrf/gr-limesdr
-> cd gr-limesdr
-> git checkout gr-3.8
-> mkdir build
-> cd build
-> cmake ..
-> make
-> sudo make install
-> sudo ldconfig
```

Figure 41: SDR Software Installation Script

Note in figure 41, that there are still many discrepancies and uncertain steps. For example, the very first installation, ([1], Figure 41) LimeSuiteGUI, requires Ubuntu version from 18.04 and above. Also note, the the GNU Radio installation ([2], Figure 41) must be 3.8.2.0 for the next step to work. Finally step 3([3], Figure 41), is the most difficult. The GR-LimeSDR Plugin is pivotal to making the system work, but it is also the most difficult to get working. This is because the plugin is made by people like us who want these two very specific components to work in very specific ways (GNU Radio and the LimeSDR). While the LimeSuiteGUI and the GNU Radio installations are both PPA or pre-packaged installations, meaning they have already been compiled and are ready to be downloaded and run on your machine, the GR-LimeSDR plugin installation is not. Instead, it is a source code installation meaning the subsequent installation script found in figure 41 is the process of installing the program files into a chosen directory/folder and then compiling the program via a make file (A file storing the necessary commands to 'make' executable programs). As a result, these files are often harder to get rid of in case you make a mistake upon installation and are generally harder to update since they are a single version/instance of that program. This installation has been used with Ubuntu 20.04.

11.3 Coarse Acquisition

Author: Ian Thomas

Coarse acquisition works by correlating the received signal with a locally generated CDMA code. This process is repeated over and over, shifting one of the codes over by one bit at a time, and picking the highest correlation value to be the CDMA *code delay*. A plot of this correlation vs. bit shift, or the circular cross correlation function, is shown below:

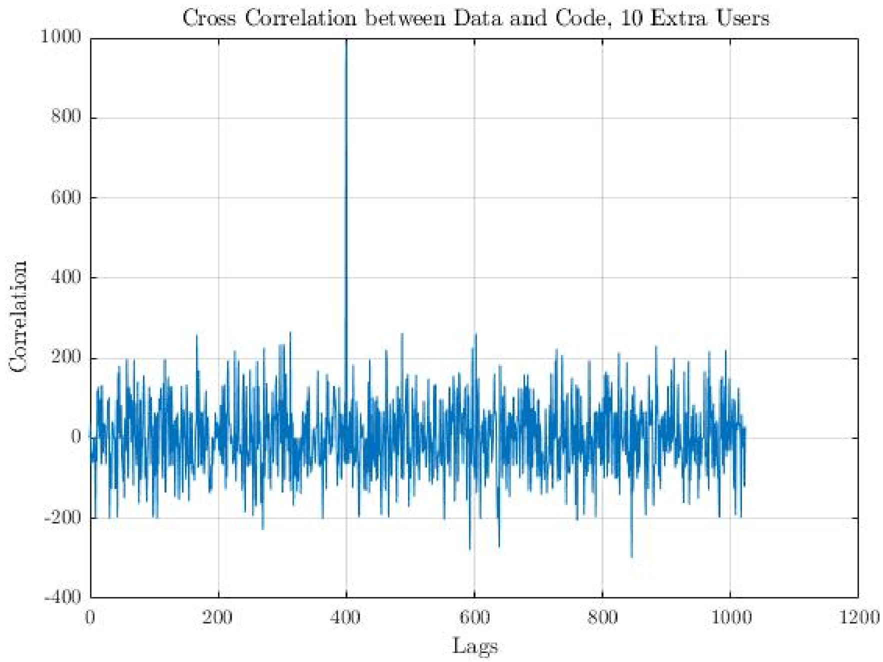

Figure 42: Circular cross correlation function of CDMA code with multiple users

There is a strong peak at a bit shift of 400, so if we shift the code 400 bits relative to the incoming data, the locally generated code and the code present in the data should be aligned and there will be a high correlation value.

This code delay is how we determine the incoming code phase, and how we can start the tracking loop to get an even finer time measurement. It is also important to determine the code phase in order to extract the data bits. The data extraction happens by correlating the local CDMA code with the incoming signal at the right delay and determining if the code is positively or negatively correlated. Positive correlation means a 1, and negative correlation means a 0.

11.4 Multi-Access Interference and Near-Far Problem

Author: Ian Thomas

CDMA operates by masking each signal with a spreading code with very low cross interference with itself and other CDMA spreading codes, but this interference is nonzero. If multiple users attempt to access the network, they will appear as white noise at a low power level, which affects the SNR and the acquisition feasibility of the signal. The SNR and acquisition feasibility can be improved with signal tracking, but the following graph shows the worst-case acquisition feasibility vs. number of users accessing the network:

A related issue is that of the near-far problem. Imagine two transmitters transmitting the same power level, but one is twice as far away from the receiver than the other. This means the signal, due to space loss, of the closer transmitter will be four times stronger than that of the further transmitter. If we are trying to acquire the further signal, we will note that the SNR will decrease by a factor of 4 (if there are no outside interferences) if the further transmitter is twice as far away. The following graph shows, again, the worst-case acquisition feasibility vs. the ratio of distances between two transmitters:

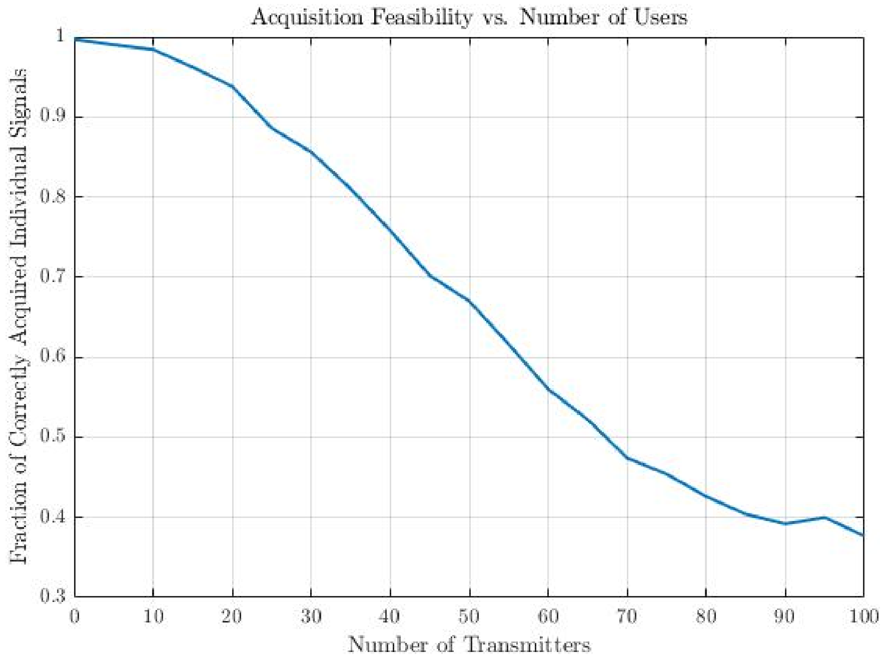

Figure 43: Multiple access interference vs. number of users

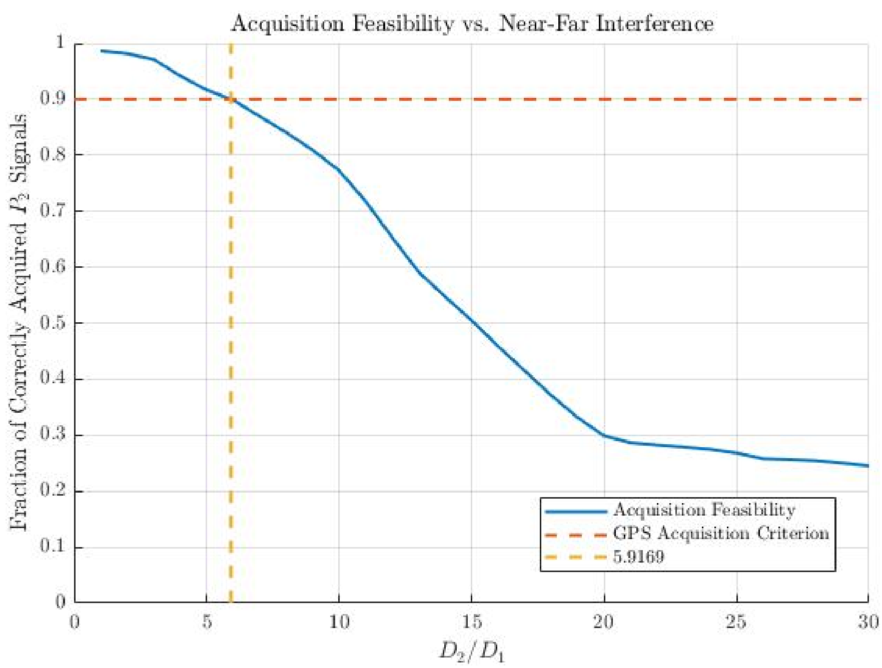

Figure 44: Near far problem vs. ratio of distances between transmitters

11.5 Dilution of Precision

Author: Fernando Palafox

Dilution of Precision (DOP) is a measure of uncertainty in a particular position solution. It quantifies error propagation as a direct effect of the satellite (or in this case, pseudolite) geometry. The closer DOP is to a value of 1, the better the geometry and the lower the error due to the geometry itself. A better geometry can be qualitatively described as one in which the satellites are well distributed around the receiver. This prevents any errors or uncertainties from a particular satellite's measurement to propagate into the position solution by leveraging the difference in position of the rest of the satellites. Figure 45 shows a visualization of this. Given how only horizontal (x and y) coordinates will be determined in P4LO's positioning system, the most relevant DOP metric is Horizontal DOP - also known as HDOP. Figure 46 shows the distribution of HDOP values within the pseudolite geometry shown in figure 7. Given the well-distributed pseudolite geometry, the system exhibits a very low overall HDOP, with no points exceeding an HDOP of 1.9.

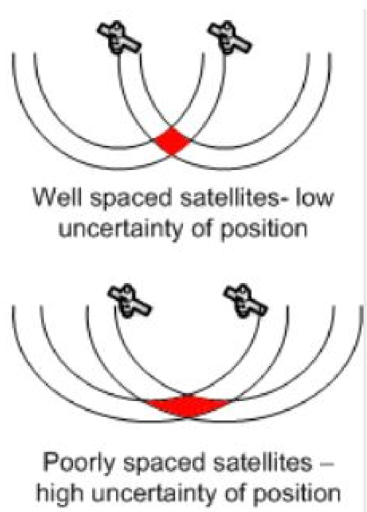

Figure 45: Visualization of uncertainty due to poor geometry

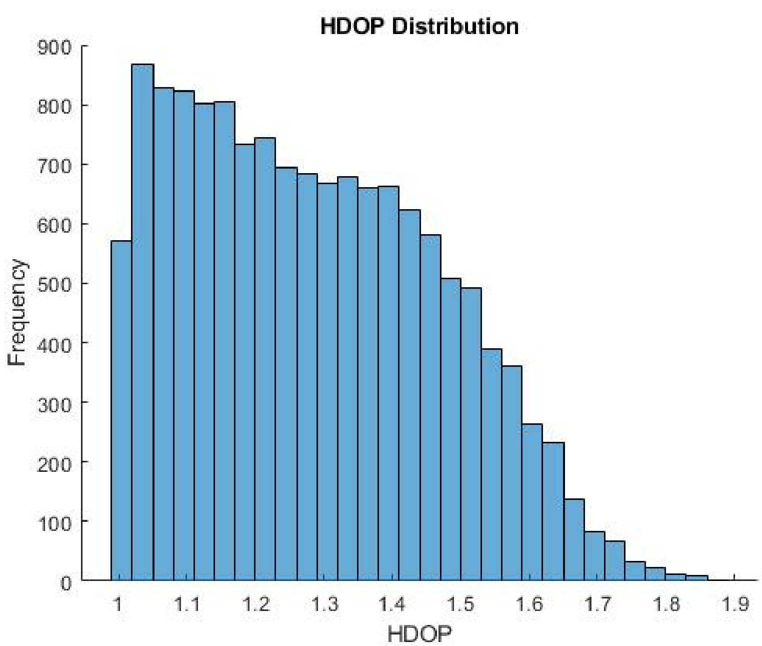

Figure 46: Distribution of HDOP values within pseudolite geometry

11.6 Positioning

Author: Fernando Palafox, Alex Lowry, Sam D'Souza

11.6.1 One-Way Ranging

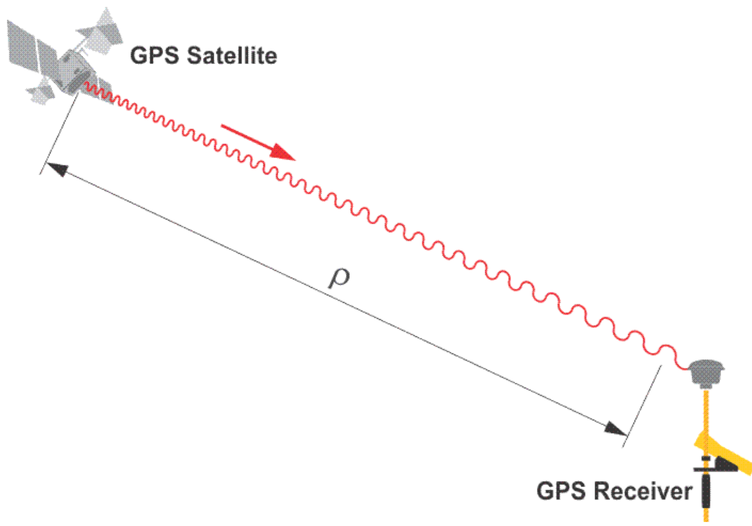

Figure 47: One-Way Ranging

11.6.2 Two-Way Ranging

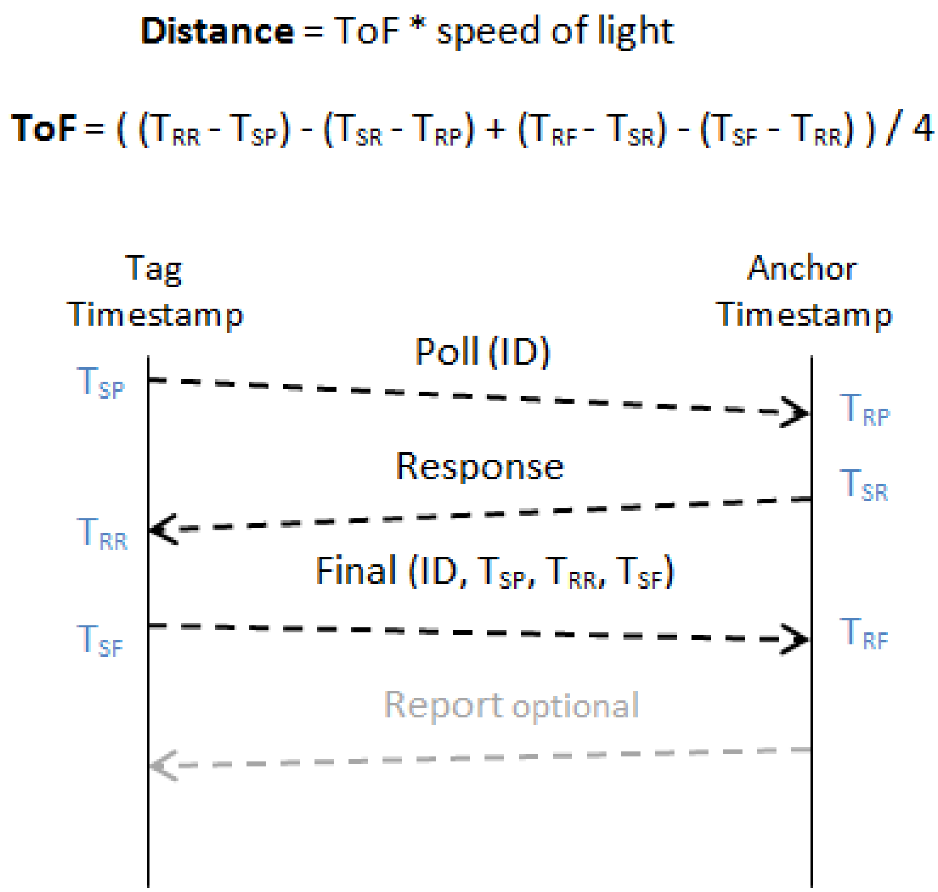

Figure 48: Two-Way Ranging

11.6.3 Time Difference of Arrival (TDOA)

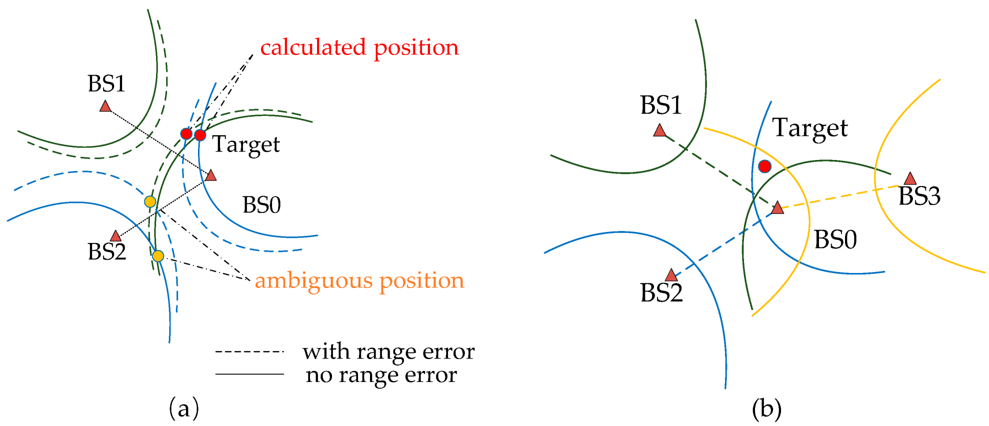

Figure 49: Time Difference of Arrival (TDOA)

11.6.4 Angle of Arrival

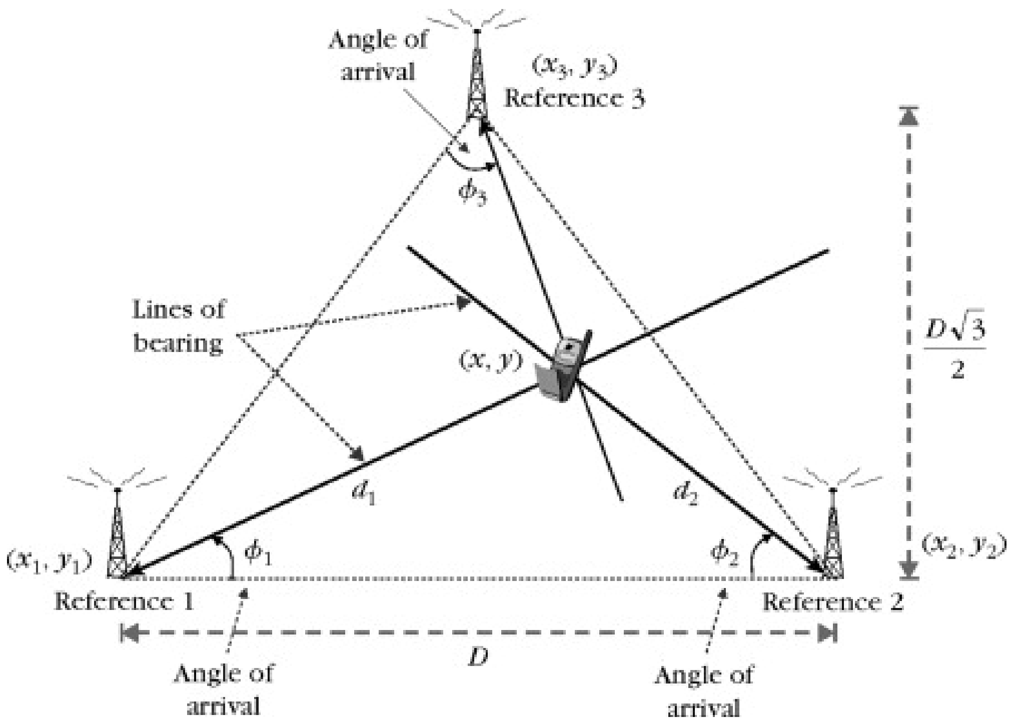

Figure 50: Angle of Arrival

11.6.5 Received Signal Strength

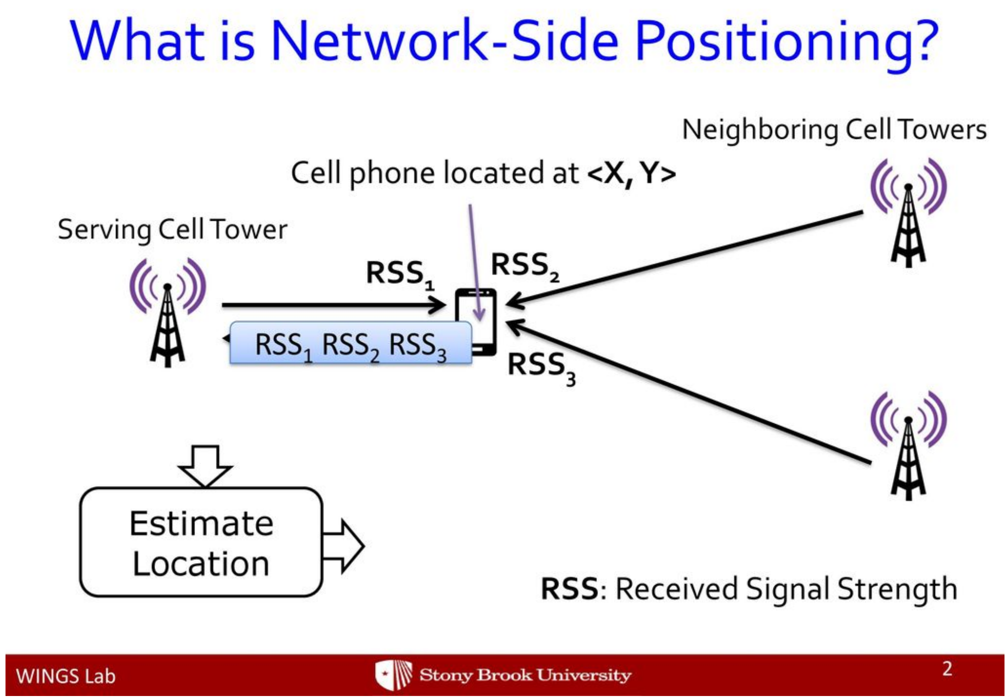

Figure 51: Received Signal Strength

12 Trade Studies

12.1 Conceptual Design Alternatives: Communications

Author: Ian Thomas

When selecting a communication link, many options must be considered. A good communication link must first and foremost satisfy any International Telecommunications Union (ITU) requirements for what is allowable in the environment we plan to design for (Lunar surface). Some additional consideration will include how this communication link might have to scale in frequency/power in order to test the functionality of the design. While the team's focus is on the handheld receiver, the team must use a communication link that is compatible with 170 simultaneous lunar surface users. Also, the communication link must be able to receive signals at an acceptable signal-to-noise ratio (SNR) in order to properly receive the signal and decode it. Additionally, since the receiver must be able to receive navigation messages to determine range and time, the team must design the link in order to accommodate both the navigation and SMS-like messages while ensuring only the intended user gets their message. The communication link can be broken into two main categories: modulation and multiplexing.

Modulation

Modulation is the process of converting an analog or digital signal into a carrier signal which is then used for transmission. A figure describing two types of analog modulation and one type of digital modulation is shown below:

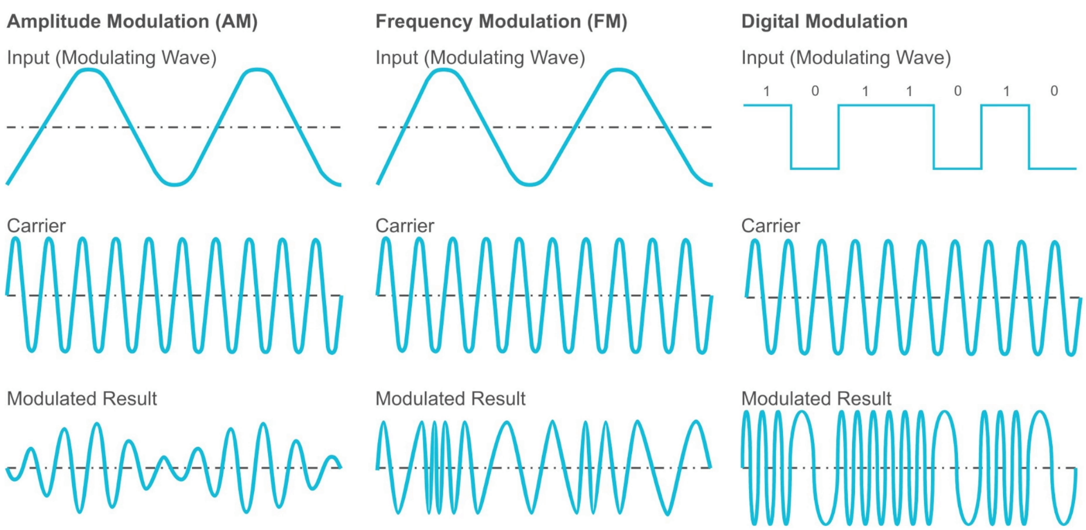

Figure 52: AM, FM, and Digital Frequency Modulation [11]

There are three main groups of digital modulation schemes: amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK) [12]. Within each modulation scheme, there are many possible ways to encode bits (or bits per symbol), which can increase the bit rate at the expense of increasing the bit error rate as signal constellation points get closer together. Additional complexity arises when combining different modulation schemes, such as quadrature amplitude modulation (QAM), amplitude phase shift keying (APSK), frequency phase shift keying (FPSK), etc. Each modulation scheme also reacts differently to different forms of interference, noise, or attenuation such as multipath, path loss, or the presence of other signals. The choice of which modulation scheme to use is also partially driven by a multiplexing scheme (for example, frequency modulation of frequency division multiplexed signals would result in severe signal interference). An in depth description of all the modulation schemes considered can be found in Appendix 11.1.1.

12.1.1 Modulation Trade Study

Table 3: Metrics and Weights - Modulation Scheme Selection

Metric	Weight	Requirement	Description and Rationale
Complexity	0.1	FR1	Ease of implementation affects how quickly we can get a solution up and running in our time frame
Bit Error Rate	0.35	3.3.1	The bit error rate is closely tied to the SNR of the signal, which affects the position/timing accuracy
Navigation Environment Viability	0.35	3.1	Will the signal actually work with Doppler, multipath, interference, and noise present on the Lunar environment
Data Rate	0.2	FR2	Will the data rate satisfy costumer requirements? Although this requirement will not make or break the project as a whole, if not satisfied, it might greatly limit some of the desired functionalities.

Table 4: Metric Values - Modulation Scheme Selection

Metric	1	2	3
Complexity	Hard to implement. Not much documentation on similar projects/implementations available in online and/or physical sources.	Challenging to implement and some documentation available.	Relatively easy to implement. Documentation for similar projects/implementations readily available in online and/or physical sources.
Bit Error Rate	High bit error rate, with respect to the other modulation schemes.	Medium bit error rate, with respect to the other modulation schemes.	Low bit error rate, with respect to the other modulation schemes.
Navigation Environment Viability	Very easily affected by environmental effects such as Doppler shift, multipath, signal interference, noise. Or, effects are NOT easily managed and/or corrected for.	Not easily affected by environmental effects such as Doppler shift, multipath, signal interference, noise.	Almost impervious to environmental effects such as Doppler shift, multipath, signal interference, noise. Or, effects are easily managed and/or corrected for.

Table 5: Trade Study Results - Modulation Scheme Selection

Metrics		n-ASK	n-FSK	n-PSK	n-QAM
Metric	**Weight**	**Score**	**Score**	**Score**	**Score**
Complexity	0.1	3	3	2	1
Bit Error Rate	0.35	2	2	3	2
Navigation Environment Viability	0.35	2	1	3	2
Data Rate	0.2	1	1	2	3
Weighted Total	1	1.90	1.55	2.70	2.10

Justification
The following list describes each modulation scheme with regards to the trade metrics:

- n-ASK
  - *Complexity* - ASK is typically quite easy to implement since all that's needed to modulate the signal is amplification.
  - *BER* - Due to ASK's closer constellation points and being amplitude modulated, noise and interference leads to a higher bit error rate.
  - *NEV* - Amplifier nonlinearities (especially on a handheld device) can cause a shift in constellation points as a function of amplitude, which would degrade the quality of the received signal [12].
  - *Data Rate* - Data rate is a function of the number of constellation points, so n-ASK could theoretically have a comparable data rate to other modulation schemes but it is limited by the constellation points being on one ray from the origin (see 11.1.1: Figs. 32, 33).

- n-FSK
  - *Complexity* - Like ASK, FSK is also quite easy to implement .
  - *BER* - FSK and ASK have comparable bit error rates [15].
  - *NEV* - FSK is heavily degraded by the Doppler shift due to the high relative velocities between moving satellites and lunar surface users.
  - *Data Rate* - The data rate is heavily influenced by how many frequencies are available within a certain bandwidth, and if guard band frequencies will be used to lower the bit error rate the number of available frequencies will be limited [12].

- n-PSK
  - *Complexity* - n-PSK is more complicated to implement because of the phase determination algorithm on the receive side. QPSK is more complicated than BPSK due to there being 4 symbols instead of 2, but both are very widely implemented and well documented.
  - *BER* - BPSK and QPSK have the same incredibly low bit error rates [15] .
  - *NEV* - BPSK and QPSK are currently in use on most Earth-based GPS systems for their robustness in a space environment and good interference/noise characteristics. They are also not as heavily affected by the Doppler shift as FSK.
  - *Data Rate* - QPSK has the double the data rate as BPSK for the same bit error rate [15], and there is no need to increase the data rate beyond that since only SMS-like data rates are desired.

- n-QAM
  - *Complexity* - QAM has the most complexity of the systems due to the combination of amplitude and phase modulation.
  - *BER* - The bit error rate is comparable to that of ASK due to the presence of the amplitude modulation within QAM, and the phase modulation doesn't contribute nearly as much to the bit error rate.
  - *NEV* - Along with ASK, amplifier nonlinearities and poor interference/noise characteristics contribute to the degradation of a QAM signal more than FSK or PSK [12].
  - *Data Rate* - The largest benefit of QAM is its incredibly high data rates — the lowest form of QAM is 16QAM which provides 4 bits per symbol (QPSK provides only 2 bits per symbol). [12]

Multiplexing Schemes
Multiplexing is the process of transmitting multiple signals simultaneously across the same medium. This is useful for this project because there are a number of different pseudolites communicating to a number of different users across one medium, and it's important to keep signals distinct from one another in order to properly

keep track of who sent them. There are three main groups of digital multiplexing schemes: time division multiplexing/multiple access (TDM/TDMA), frequency division multiplexing (FDM/FDMA), and code division multiplexing (CDM/CDMA). Each employ different strategies in order to send multiple signals across a shared medium, and often times, the strategies are used together to further diversify and optimize multiplexing schemes to meet requirements. The three main multiplexing schemes considered are time division multiple access (TDMA), frequency division multiple access (FDMA), and code division multiple access (CDMA). An in depth description of all the multiplexing schemes considered can be found in Appendix 11.1.2.

12.1.2 Multiplexing Trade Study

Table 6: Metrics and Weights - Multiplexing Scheme Selection

Metric	Weight	Requirement	Description and Rationale
Complexity	0.2	FR1	Ease of implementation affects how quickly we can get a solution up and running in our time frame
Number of Users	0.2	FR6	The multiple access scheme must be able to support the number of users required by the customer
Bit Error Rate	0.3	3.3.1	The BER is closely tied to the SNR of the signal, which affects the position/timing accuracy
Navigation Environment Viability	0.3	3.1	Will the signal actually work with Doppler, multipath, interference, and noise present on the Lunar environment

Table 7: Metric Values - Multiplexing Scheme Selection

Metric	1	2	3
Complexity	Hard to implement. Not much documentation on similar projects/implementations available in online and/or physical sources.	Challenging to implement and some documentation available.	Relatively easy to implement. Documentation for similar projects/implementations readily available in online and/or physical sources.
Number of Users	Does not meet number of user requirement.	Meets number of users requirement.	Meets or exceeds number of user requirement.
Bit Error Rate	High bit error rate, with respect to the other modulation schemes.	Medium bit error rate, with respect to the other modulation schemes.	Low bit error rate, with respect to the other modulation schemes.
Navigation Environment Viability	Very easily affected by environmental effects such as Doppler shift, multipath, signal interference, noise. Or, effects are NOT easily managed and/or corrected for.	Not easily affected by environmental effects such as Doppler shift, multipath, signal interference, noise.	Almost impervious to environmental effects such as Doppler shift, multipath, signal interference, noise. Or, effects are easily managed and/or corrected for.

Table 8: Trade Study Results - Multiplexing Scheme Selection

Metrics		CDMA	TDMA	(O)FDMA
Metric	**Weight**	**Score**	**Score**	**Score**
Complexity	0.2	2	1	1
Number of Users	0.2	2	1	3
Bit Error Rate	0.3	2	1	3
Navigation Environment Viability	0.3	3	2	1
Weighted Total	1	2.3	1.3	2.0

Justification

- CDMA
  - *Complexity* - the complexity for CDMA lies in decoding the signal — a receiver with the same spreading code has to match the code phase exactly with the receive signal in order to properly decode the signal. Beyond the phase matching there isn't much additional complexity, and CDMA systems for ranging and timing are well documented because of its widespread use in Earth-based GPS systems.
  - *Number of Users* - the number of users of a CDMA signal is bound by the ratio of the spreading code bit rate to the data signal bit rate, called the spreading factor, and there can't be any more users on the system than this upper limit.
  - *BER* - the bit error rate for CDMA largely has to do with which kind of code is used to spread the signal. The best codes for this job are pseudo-random Gold Codes [26].
  - *NEV* - CDMA is widely used in Earth-based GPS systems because of its ability to provide utility in the presence of Doppler shift, multipath interference, and its low inter-signal interference due to CDMA codes typically looking like noise without the right code
- (O)FDMA
  - *Complexity* - Regular FDMA is typically easy to implement, but the bandwidth restriction means some sort of OFDMA must be implemented. The frequency control to get enough bands to make OFDMA viable in a narrow bandwidth increases the complexity significantly [17].
  - *Number of Users* - The number of users in an OFDMA system is quite high barring the complexity barrier, since a high number of closely spaced orthogonal frequencies can fit even in a narrow band, provided the channel environment allows for many closely spaced frequencies
  - *BER* - OFDMA is even more resistant to Gaussian white noise than CDMA [17], which means its bit error rate is quite low provided the channel environment allows for OFDMA to operate well
  - *NEV* - Unfortunately, the presence of a Doppler shift due to the high relative velocity between satellites and lunar surface users means that the tightly spaced frequencies required for OFDMA to work will be distorted
- TDMA
  - *Complexity* - TDMA is simple in concept, but synchronizing the sender and the receiver to the precise interval of the signal transitions requires very precise timing for this technique to be effective [16].
  - *Number of Users* - The number of users is highly limited by the data rate of each user and how precise the timing is between sender and receiver. Often times guard time intervals must be included in the signal to account for imprecise timing, which comes at the cost of the number of users [16].
  - *BER* - As mentioned, guard intervals must be included in the signal to account for any timing imprecision between sender and receiver. This is to account for the large problem of bit errors if the receiver time isn't precisely synchronized with the transmitter time.
  - *NEV* - Other than the precise time synchronization, TDMA signals don't suffer Doppler shift nearly as much as FDMA does.

12.2 Conceptual Design Alternatives: SDR Software Packages

Author: Ruben Hinojosa Torres, Brendan Palmer

One of the main issues with the transmission and reception of electromagnetic signals via hard-wired electronics is the flexibility in which these systems can provide. The team desires a flexible testing/simulation environment in which only one electronic component is used to transmit and receive signals consisting of all sorts of coding and decoding schemes. This is where the LimeSDR comes into play. This electronic device, as the name implies, is a software defined radio in which any signal can be received (or transmitted) in the various information schemes currently in use. These SDRs allows us to manipulate this signal via software. This is more beneficial, since the software previously used was a set of electronic components and circuits tasked with processing the incoming or outgoing signal. Naturally, this new approach results in added complexity on the software side, since now, most of these components need to be converted into software. However, this software has multiple solutions which engineers have developed throughout the years. Because of this, a decision must be reached as to which piece of software is best suited for this project. This aspect of the project engulfs a large part of our design considering most of the testing and development will occur within the capability of our selected software package. To be specific, the objectives influenced by the selection of a software package in our project are found in Section 2: Levels 1, 2, 3, 4, and 6.

For all three of our software package candidates, we can devise a ranking system such that the most suitable candidate is chosen among the options. This ranking system contains:

- Software Complexity: Determines the accessibility and readability of the code developed.
- Software Documentation: Considering most of our project is open source and would otherwise have to be developed over the course of years, we opted to employ the availability of open source programs.
- Software limits: Given some of the open source programs available are limited to their capabilities, we must employ a software package which is able to complete the tasks required by the project.
- Software Availability: This will manage whether the software is compatible with the systems (Laptop/Raspberry Pi) which will be used throughout the project.

We will dive into the pros and cons of three programs: LimeSuiteGUI and Gqrx and GNU Radio.

LimeSuiteGUI

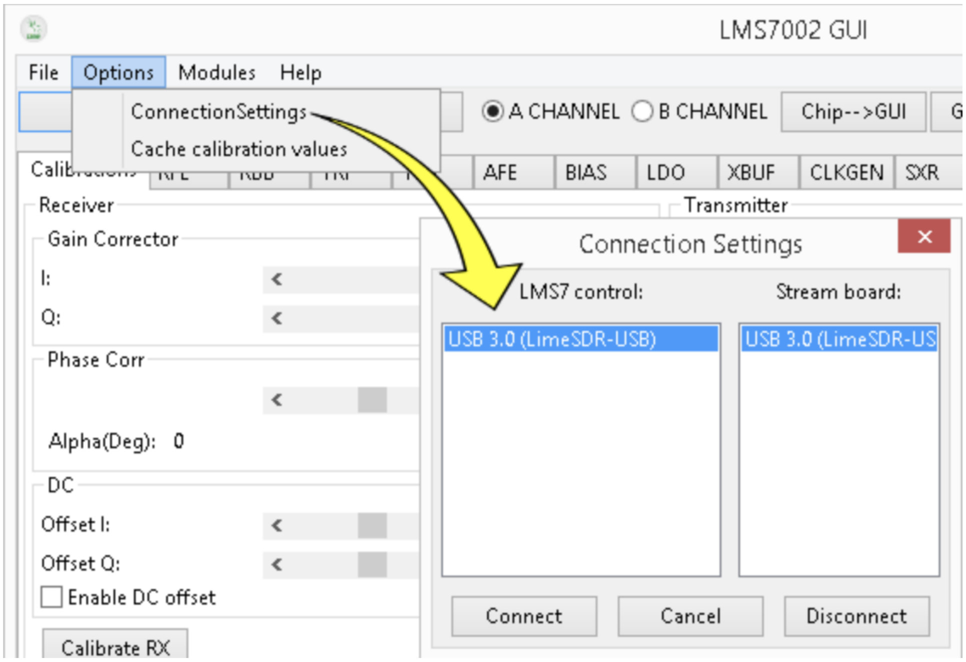

Figure 53: Example of LimeSuiteGUI control panel

Pros: Available on all platforms including Windows, Linux, and MacOS. Plenty of documentation is available for operation, though description of the software is limited. GUI should allow the project to accomplish all tasks including storing data as well as transmitting, receiving, and analyzing signals. The GUI was also developed specifically for the LimeSDR so no plug-ins are required.

Cons: Non-Windows platforms are difficult to install and the instillation of dependencies for these non-Windows platforms are cumbersome. Thus, the setup and groundwork may be more difficult for this software. GUI is less user-friendly than the alternatives. It may be more difficult to perform the simple functionalities that this project calls for.

Gqrx

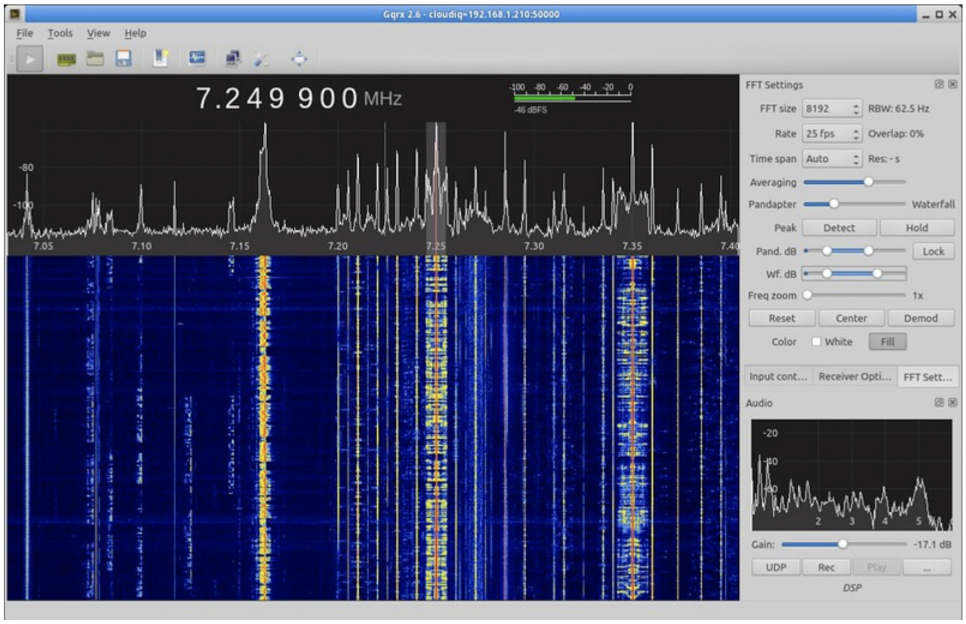

Figure 54: User interface for Gqrx

Pros: The UI is incredibly user-friendly. The download is very easy through source code. The software allows for easy control of frequency and gain. The software provides a demodulator that will fulfill all necessary requirements for the receiver. It also allows for simple saving and frequency analysis of the received signal.

Cons: The software runs on Linux and MacOS, but has no support for Windows, which may be the operating system we run on. The LimeSDR is only supported through the SoapySDR device string. It acts only as a receiver and will not help with the transmission element of our project. Some of the best documentation comes from third-party sources instead of the developer.

GNU Radio

Pros: The quantity and quality of the documentation hints at the fact that this software package will be the most helpful in terms of the learning curve that comes with any new language and environment setup. Once the GNU Radio setup is complete, it also seems like the tools which are available will greatly simplify our analysis in the sense that most of the ground up development is already done and therefore the team can focus on the higher level concepts of how each tool works. Below in figure 55, we can see just how simple creating a program would look like for our system. In figure 56, we can see the output of that example.

Cons: Because the software package is open source, a lot of time and effort will be required to get GNU Radio setup on the LimeSDR and laptop. It can also be said that because most of the component blocks have already been created, the low level information and/or circuit setup might be out of reach of the programmer.

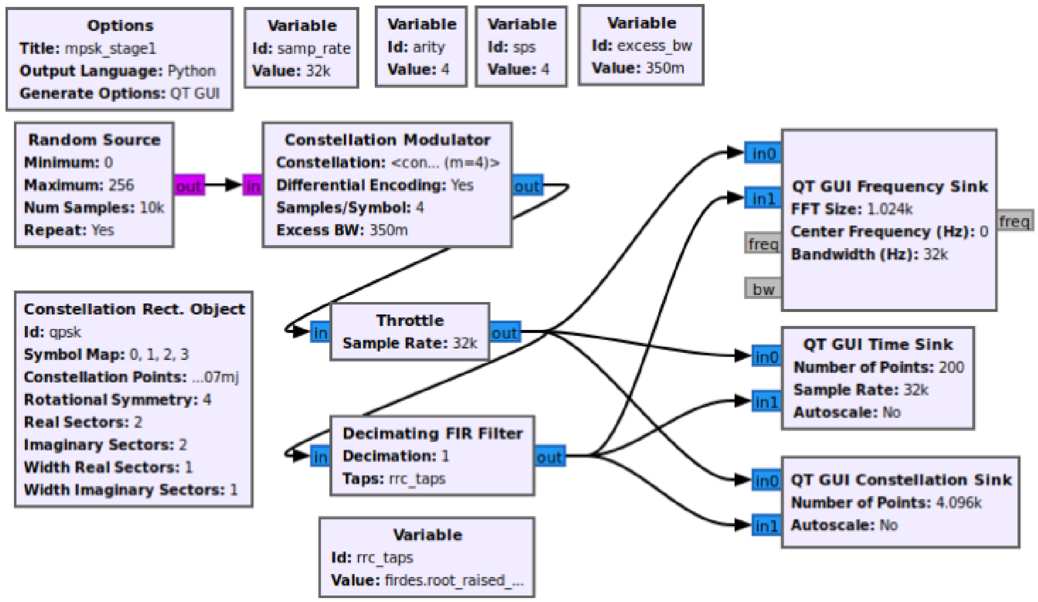

Figure 55: Example flow chart for simulation of M-PSK signal. [9]

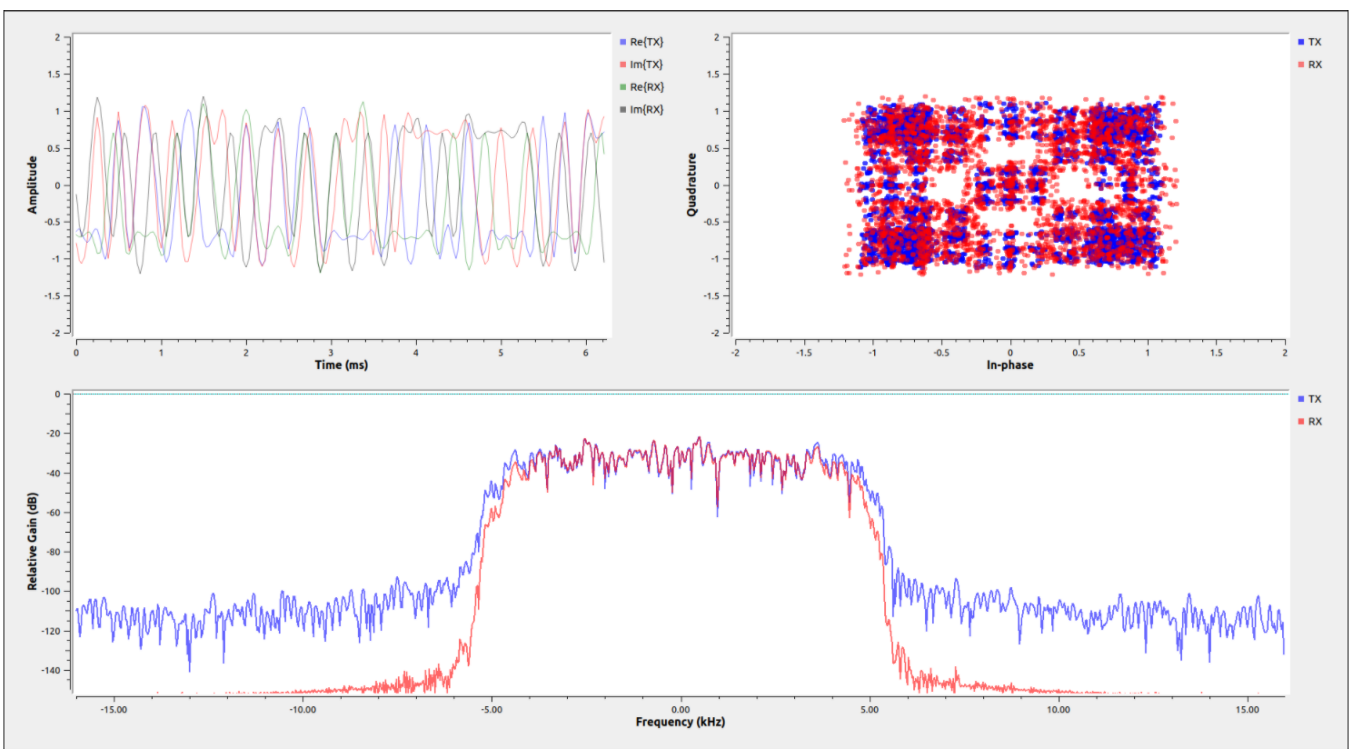

Figure 56: GNU Radio: Example simulation data for M-PSK signal. [9]

12.2.1 Trade Study: LimeSDR Software Packages

Table 9: SDR Metric Rationale

Metric	Weight	Requirement	Description/Rationale
Complexity	**25%**	**3.1.3**	In order for the system to be scaled into a working architectural model, the software used during development must be fast and easy to use. The level of complexity will influence the programmers choice of software and hence, will determine whether the software is used or not.
Documentation	**30%**	**3.1.3**	For the system to be scaled properly into a working architecture, the software must be well documented so that the programmer is able to predict/anticipate any short comings of the software.
Limits	**30%**	**3.2.1** **3.6.1**	In order for the system to demonstrate transmission and reception of data, the software must first be able to provide said capabilities to the hardware. In order for the system to handle 170 users at a time, the software must not be limited in this respect. Hence, a limitation on the number of broadcasted signals is a limitation which will determine whether the software is used or not.
Availability	**15%**	**3.1.3**	So that the system is able to be be scaled up, the underlying hardware and software must be readily available to any developmental set up. This is in regard to the many platforms widely used today.

Metric Meaning: LimeSDR Software Packages

Table 10: SDR Metric Meaning

Metric	1	2	3
Complexity	Software is time inefficient, complex, and not easy to use. Would require an extensive amount of time to get well versed.	Software is time efficient, complex, but easy to use given the amount of time for development.	Software is time efficient, not as complex, and easy to use.
Documentation	Software is not well documented.	Software is documented but lacks specificity.	Software is very well documented.
Limits	Software lacks either transmission or reception capabilities. Software will not support various users. Clearly limited in SDR capabilities.	Software can do signal reception. Software can support various users.	Software package will fulfill all given requirements.
Availability	Software only accessible for Windows based systems.	Software available for Windows and MacOS systems.	Software available for Windows, MacOS, and Linux based systems.

Final Decision: LimeSDR Software Packages

Table 11: Final Software Package Decision

		LimeSDR GUI	GQRX	GNU Radio
Metric	**Weight**	**Score**	**Score**	**Score**
Complexity	0.25	3	3	3
Documentation	0.30	1	2	2
Limits	0.30	3	1	3
Availability	0.15	1	1	2
Weighted Total		2.1	1.8	2.55

Final Decision Justification: LimeSDR Software Packages

- *GNU Radio Complexity Score: 3*

  The GNU Radio runs on flow-graph type framework and therefore is easy to understand and work with. This sort of framework also works well for scaling up into large architectural problems considering detailed flow-graphs are often used for directional control of information. The complexity of the GNU Radio also allows for quick analysis of complex systems by the grouping of components, which allows for a level of abstraction between the programmer and what they want to achieve. Apart from this, the GNU Radio offers the ability to characterize parameters for set blocks, which act as functions, or entire electrical circuits. This helps to test quantities or characteristics in a modular fashion, which again aids in the overall desired scaling of the project.

- *GNU Radio Documentation Score: 2*

  The GNU Radio has a very detailed documentation web page [7]. This again helps in the future scaling of the project. The various 'getting started' pages on the GNU website include ranks such as Beginner, Intermediate, Expert, and Developer pages. Each with its own subset of easy to read examples and 'walk-through's'; these are extremely helpful on the simulation aspect of the GNU Radio software but also in the hardware side considering many of the problems with hardware can be modeled all within the software of the GNU Radio application. With these tools we will be able to provide a good prediction of what the communication will look like between our pseudolites and users.

- *GNU Radio Limits Score: 3*

  Reviewing some of the examples in the documentation page of the GNU Radio, it's clear that the GNU Radio is a great tool for the simulation and employment of our project. With the given blocks (Modulation Scheme blocks, Filter blocks, Instrumentation blocks, etc.) already available for use and modification, we can build the system needed for an SDR which can transmit and receive in various signal quantities and schemes.

- *GNU Radio Availability Score: 2*

  Given the Gr-limeSDR Plugin for GNU Radio, we can see that this requirement is fulfilled in all three desired platforms: Windows, Linux, and MacOS. This is important for the project's architectural scalability and ease of access.

12.3 Conceptual Design Alternatives: Antennas

Author: Ventura Morales

Another critical element of this project is the antenna. Based on customer needs, the antenna has to satisfy two functional requirements: transmit signals at 2.4 GHz and receive at 2.48 GHz carrier frequency. With this in mind, bandwidth (frequency range) wasn't the only factor taken into consideration when selecting the antenna. Factors such as weight, size, performance, cost, gain and compatibility played a role in the final decision. Before going more in-depth on the different antenna configurations, different antenna configurations were analyzed. Three were considered: externally mounted, attached, and embedded. These can be seen in the following figures: 57, 58, 59.

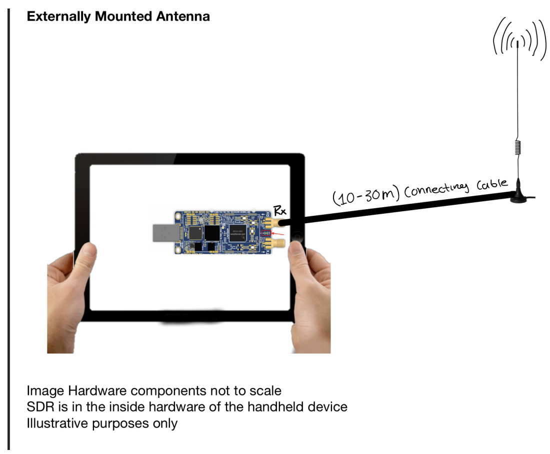

Figure 57: Externally Mounted Antenna Configuration

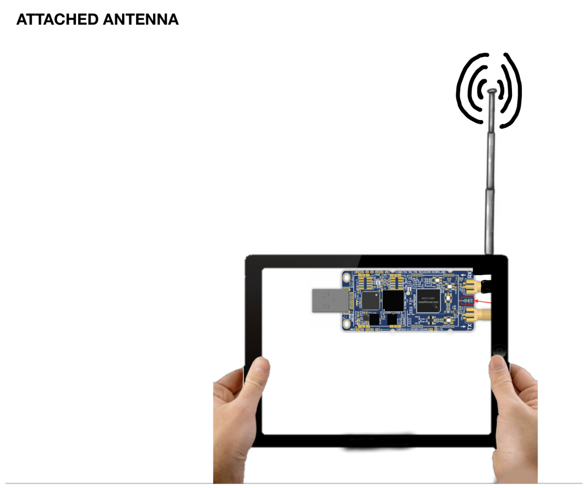

Figure 58: Attached Antenna Configuration

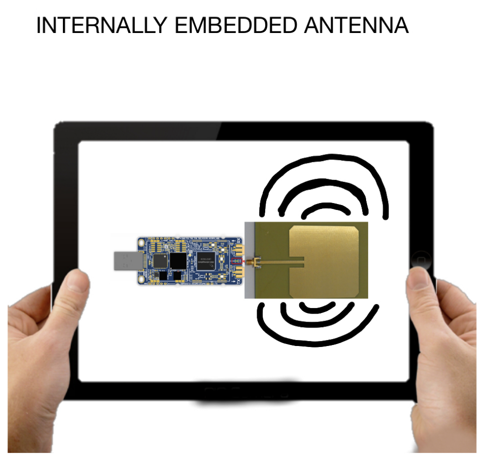

Figure 59: Embedded Antenna Configuration

Sequentially, different antenna candidates were selected. The search started with previous teams and advisors recommending websites, antenna companies, and team personal research on different providers. For example: Spark-fun, Laird, and Fairview were primarily used to search for ideal antenna candidates.

After many options were considered, the following four candidates were selected: TE Connectivity Antenna, Laird-MAF94051, Laird OC24006H and Argain-N2420M. All of these candidates satisfied the main functional requirements, but further analysis was done to down-select to the final choice. Trade matrices were designed to accommodate customer requirements, but also to satisfy other considerations described at the beginning of this section (size, cost, weight, etc.) The antenna trade matrix can be seen in figure 60.

Metric	High Score (3)	Medium Score(2)	Low Score(1)
Cost Effectiveness	low cost for a high performance and capabilities (price<5)	The antenna has an average cost for its capabilities(5-30)	The antenna has a high cost for its capabilities (Over 30)
Weight	The antenna is very light (Under 50 grams)	The antenna has a medium weight (from 50-200grams)	The antennas has a high weight value (over 200 grams)
Compatibility	The antenna can easily connect to the SDR, The user has no issues when using the antenna	The antenna is compatible with the sdr without any extra hardware but needs intervention for it to start working	The antenna is not compatible with the sdr without any extra hardware and needs intervention for it to start functioning
Performance/Specs	Omnidirectional, Operates in designated frequency, high bandwidth, good materials, Large Effective Area,etc.	The antenna has all the required specs but does not have good materials or other hardware constraints	The antenna doesn't satisfy all the needed requirements

Figure 60: Antenna Trade Matrix

Next, we have the previous mentioned antennas, with their general performance specifications and characteristics. These can be seen in figures 61 and 62.

Antenna Model	Antenna Type	Antenna Design Configuration	Gain (dBi)	Bandwidth(GHz)
TE Connectivity Antenna	Omnidirectional-DualBand	Embedded	2	2.4-3.8 GHz, 5.150-5.870 GHz
Laird-MAF94051	External Dual-Band Omnidirectional	Attached	2	2.4-2.5
Laird OC24006H	Omnidirectional/ horizontally polarized	External	6	2.4-2.5
Argain-N2420M	Single Band embedded	Embedded	2.81	2.4-2.49

Figure 61: Antenna Performance Table

Antenna Model	Weight (grams)	Unit-Cost (USD Currency)	Connector Type	Effective Area (Ae) or Effective Antenna Aperture (m^2)
TE Connectivity	3.3	4.79	Cable-Side Entry	0.001828
Laird-MAF94051	113.4	7.64	RP SMA Connector with 90 degree elbow	0.001814
Laird OC24006H	260	50.20	N-Female	0.004556
Argain-N2420M	0.5	1.80	IPEX/MHF/U FL	0.002186

Figure 62: Antenna Specs Table

To add more context to the previous tables, lets explain the definition of gain and effective area. Antenna gain is a performance factor that measures electrical efficiency and wave directivity. For example, if an antenna transmits a signal, gain is a measure of how well the antenna's inputted power is transformed into RF wave signals, as well as measuring the directivity (direction of radiated power waves). A high gain antenna (HGA) radiates most of the power in a particular direction but in exchange, reduces vertical propagation (The Sphere Effect). Low gain antennas (LGA) have omnidirectional wave propagation but not as much range when compared to a HGA. A graph of this property is shown below in figure 63

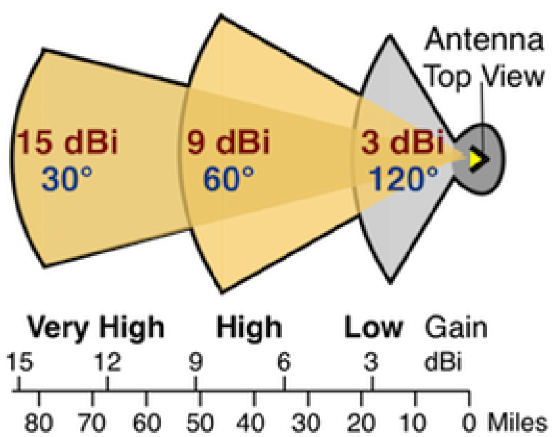

Figure 63: High and Low Gain Antenna Comparison

Finally, the antenna's effective area (also called effective aperture) is a measure of how efficiently an antenna receives power. The effective area equation is as follows:

$$A = \frac{G}{4\pi} * \frac{c^2}{f^2} \tag{1}$$

where G is the antenna gain, c is the speed of light and f is the used frequency.

Moving forward, while taking into account the various technical factors and other constraints, the team decided to rate the antennas based on the trade matrices to come to a final design choice. Based on the trade matrix scores seen in figure 64, the selected candidate is the Dual band TE connectivity antenna (TE). This candidate outperformed all other options in every category, while the provider (Spark-fun) has a sufficient amount of technical documentation, despite being a low cost antenna. The TE antenna is displayed in Section 6.4: Detailed Design: Antennas, Fig. **??**.

Metric	Weighting	Antenna #1 TE	Antenna #2 Laird-1	Antenna #3 Laird-2	Antenna #4 Argain
Cost	0.15	3	2	1	3
Weight	0.30	3	2	1	3
Compatibility	0.15	3	3	2	3
Performance/Specs	0.40	3	3	3	2
Total Score	100%	100%	80%	65%	90%

Figure 64: Point Trade Matrix Results

12.4 Conceptual Design Alternatives: Positioning Algorithm

Author: Fernando Palafox

A crucial element of P4LO is the positioning system used to generate a position solution. Although there are many possible approaches to develop this, the team ultimately decided to go for trilateration with one-way ranging. This is the same algorithm used in modern global navigation satellite systems such as GPS, GLONASS and Galileo. In order to decide on this algorithm, a trade study was conducted in which each of the positioning algorithms was scored using a set of 3 weighted metrics: performance, complexity, and documentation. Descriptions for each of the studied positioning algorithms can be found below. A detailed description of each of these metrics, their corresponding functional/design requirements, weights and rationale can also be found below. Each of the metrics can have a score ranging from 1-3, and each of these levels is also explained in detail.

Trilateration and one-way ranging

In this method, a network of satellites is set up to transmit signals which contain encoded navigation messages. These navigation messages contain information about the satellite position as well as the specific time when the messages were sent (time of transmission). The receiver takes note of when the signal was received (time of reception) and uses the known signal velocity (speed of light) to calculate the distance from the transmitter to the receiver in a process known as ranging. Once ranges for at least 4 receivers have been calculated, the receiver uses the satellite position data encoded into the signal to determine exactly where the satellites were when their messages were sent. With this information, a system of equations with four unknowns can be set up and solved for: x, y, z and receiver clock error. This method requires extremely accurate and stable clocks on-board the transmitting satellites, otherwise the system of equations breaks down. The one-way nature of the algorithm (no uplink to the satellites, only downlink) and the fact that receivers do not need very accurate clocks (and can therefore be relatively simple and cheap devices) means the system readily lends itself to be used by a great number of users. This is the method currently used in all modern global navigation satellite systems such as GPS, GLONASS and Galileo and there is therefore a trove of information available on how it works. Furthermore, CU Boulder has plenty of faculty members with specific experience on this positioning algorithm (Professor Morton and Professor Akos to name a couple). An image showing this positioning system is shown in Appendix 11.6 figure, 47.

Trilateration and two-way ranging

This algorithm uses a variation of the ranging process in which range is determined through a back and forth communication between the transmitters and receivers. The advantage of this method is that it does not require the transmitters to have very accurate clocks which reduces costs and maintenance complexity on the transmitter side. However, this does mean that an uplink transmission must be setup between the receiver and the satellites in order to correctly compute ranges. This limits the number of users and adds a a lot of complexity to the implementation. An image showing this positioning system is shown in Appendix 11.6 figure, 48.

Hyperbolic positioning and Time Difference of Arrival (TDOA)

Two synchronized transmitters send out signals at the same time. The location of these transmitters is known in advance. The receiver then measures the differences between range measurements of the two transmitters and generates a hyperbola. This same process is repeated with another pair of transmitters whose location is also known in advance. The intersections of these two hyperbolas denotes the location of the receiver and any ambiguities can be solved by a rough apriori estimate of the receiver location or by conducting additional measurements. This system was used in the radio navigation systems of Loran and Omega and is also known as a time difference of arrival (TDOA) system. Advantages of this system include the fact that the receiver clock does not need to be very accurate. Disadvantages are that the receiver clocks do need to be extremely accurate and that some kind of outside knowledge is needed in order to solve the hyperbolic intersection ambiguities. An image showing this positioning system is shown in Appendix 11.6 figure, 49.

Angle of arrival (AoA) and one-way ranging

In this system three transmitters transmit unique signals which a receiver can tell apart. The receiver then takes this signal and using a variety of methods such as an antenna array, determines the angle at which it's receiving the signals from each of the transmitters. Using this angle information as well as the transmitter positions, it can generate a system of equations to calculate position. An advantage of this method is that it does not require synchronized clocks on either the receiver or the transmitters. This greatly reduces complexity on the timing side of things. However, this algorithm requires a receiver with an antenna capable of determining angle of arrival which can be extremely complicated and expensive on both the hardware and software sides. Although some angle of arrival principles are used in cellular networks, documentation is not as readily available as it is for other more popular radio navigation methods. An image showing this positioning system is shown in Appendix 11.6 figure, 50.

Received Signal Strength (RSS)

This algorithm uses at least three transmitters. Each of these is at a known location and transmits at a known power. The receiver listens to each of the transmitted signals and notes its power. Then, using the inverse-square relationship between power and distance, it can calculate the range to each receiver. Once it has these ranges, it can generate a system of equations to calculate its exact position. The advantage of this system is that it does not require expensive, stable and accurate clocks on either the receiver or the transmitter. However, it does require a very delicate sensing system on the receiver that can determine tiny variations in signal strength which means that the system would inherently be extremely sensitive to environmental noise and interference. Furthermore, documentation is not as readily available as with other positioning algorithms. An image showing this positioning system is shown in Appendix 11.6 figure, 51.

12.4.1 Positioning Algorithm Trade Study

Table 12: Metrics and Weights - Positioning Algorithm

Metric	Weight	Requirement	Description and Rationale
Performance	0.4	FR1, FR3, FR6	How well the positioning algorithm will be able to calculate a position solution that aligns with the requirements set in FR3
Complexity	0.3	FR1	Ease of implementation and how quickly we can get a solution up and running in our time frame
Documentation	0.3	FR1	How much documentation is available in sources such as textbooks, internet, research papers, etc...

Table 13: Metric Values - Positioning Algorithm

Metric	1	2	3
Performance	Does not meet many of the requirements set by FR1, FR3, FR6.	Meets most of the requirements set by FR1, FR3, FR6.	Meets all of the requirements set by FR1, FR3, FR6.
Complexity	Algorithm is hard to implement and very complex. Requires many moving different pieces to work flawlessly or else it will not work well.	Algorithm is complicated and complex but certainly within our implementation capabilities for the project time frame.	Algorithm is straightforward and relatively easy to implement.
Documentation	Very little documentation available. Would require considerable time and effort to find all the necessary information to implement this algorithm - perhaps more than it's worth.	Good amount of documentation available but not many people close to us knowledgeable on the subject. May pose some difficulties if we have questions that aren't answered in the existing documentation.	Ample documentation available in sources such as books, research papers and the Internet. Furthermore, we have access to people with this knowledge such as faculty and/or professors.

Table 14: Trade Study Results - Positioning Algorithm

Metrics		**TOA, 1**	**TOA, 2**	**TDOA**	**AoA**	**RSS**
Metric	**Weight**	**Score**	**Score**	**Score**	**Score**	**Score**
Performance	0.4	3	2	2	3	2
Complexity	0.3	2	1	2	1	2
Documentation	0.3	3	2	2	1	1
Weighted Total	1	2.7	1.7	2	1.8	1.7

Justification
The following list describes each modulation scheme with regards to the trade metrics:

- Trilateration and one-way ranging (Time of Arrival, TOA 1)
  - *Performance* - This algorithm has been shown to work to the desired precision of 10m with a large number of users (see global navigation systems like GPS) . This algorithm would have no problem satisfying FR1, FR3 and FR6 and therefore receives the highest score in performance.
  - *Complexity* - Although the concept is relatively simple, this algorithm does require very accurate clocks on the transmitters. This adds an extra level of complexity which lowers the score to a 2.
  - *Documentation* - The ubiquity of global navigation systems which use this positioning algorithm means that there are many resources from which to acquire all the necessary information to implement this algorithm. Furthermore, the team has access to experts such as Dr. Jade Morton and Dr. Dennis Akos, both faculty at CU Boulder, whom have ample experience in using this positioning algorithm.
- Trilateration and two-way ranging (Time of Arrival, TOA 2)
  - *Performance* - Although this positioning system has the capability to achieve the required precision, the added complexity with back and forth communications between the transmitter and receiver greatly limits the number of users as well as the scalability of the system. However, in theory it is possible. Therefore, it is given a score of 2.
  - *Complexity* - The added level of sophistication necessary in the communications link to handle two-way ranging adds an enormous layer of complexity to the project - which would only be exacerbated

if one were to satisfy the 170 user requirement. Therefore, this algorithm receives the lowest score for complexity - a score of 1.

- *Documentation* - Although some documentation is available as two-way ranging is commonly used in home networks, the documentation and/or existing resources are not specifically geared towards implementing a scalable positioning system that would work in an environment such as the Moon. Therefore, this system receives a 2.

- Hyperbolic positioning and Time Difference of Arrival (TDOA)
  - *Performance* - Although TDOA should in theory be able to handle the user requirements, getting the necessary precision of 10m might prove to be a little more complicated given how the receiver would have to deal with intersection ambiguities.
  - *Complexity* - This system should be relatively straightfoward to implement. However, it still requires accurate synchronized clocks on both transmitters. Therefore, and as with TOA 1, it recieves a complexity score of 2.
  - *Documentation* - Given how this system was used in radio-navigation systems such as LORAN and Omega, some documentation exists on the subject. However, this algorithm is not used much these days and the documentation is therefore not very recent. Furthermore, it is geared towards more large scale systems rather than the reduced prototype the team will attempt to build.

- Angle of arrival (AoA) and one-way ranging
  - *Performance* - This algorithm should be able to satisfy all functional requirements and therefore recieves a performance score of 3.
  - *Complexity* - Antenna arrays that can accurately detect angle of arrival of a signal are not only expensive and large, but they are also very complex to operate and develop. They would require intricate studies on antenna patterns and behavior and would most certainly take longer than the year the team has to develop. Therefore, AoA receives the lowest score of 1 for complexity.
  - *Documentation* - Although Angle of Arrival is commonly used in cellular networks, there is not much documentation readily available. Furthermore, the documentation is mostly geared towards cellular networks rather than a positioning system like the one the team is attempting to develop. Finally, the team doesn't have any easy access to experts in this field which would hinder progress if any questions were to arise during the development process. Therefore, this AoA receives a 1 for documentation.

- Received Signal Strength (RSS)
  - *Performance* - Although this algorithm should be able to support the required number of users, it will struggle with achieving the necessary positioning accuracy due to its sensitivity to noise. Therefore, it receives a reduced score of 2 for performance.
  - *Complexity* - The principles that govern this algorithm are relatively simple. However, the implementation might prove to be complicated given how the team will have to gain a very precise understanding of exactly how signal strength varies with distance - particularly in a scenario with lots of noise and/or interference.
  - *Documentation* - Not much documentation is available on this front. Therefore it receives a score of 1.

After conducting the trade-study, trilateration with one-way ranging received the highest score 2.7 and was chosen as the final positioning algorithm for P4LO. This decision makes sense considering how much documentation is available and the access the team has to experts on the algorithm such as Dr. Jade Morton and Dr. Dennis Akos. Although this algorithm does have some significant technical hurdles (such as the need for accurate, stable and synchronized clocks on the receivers), they are all within reach of the team and can certainly be solved within the project time frame of two semesters.